Fingerprint Classification Techniques

by

Chainarong Klimanee, B.E.

Engineering

Submitted in fulfilment of the requirements

for the degree of

Master of Engineering Science

University of Tasmania (April, 2004)

Statement of Originality

This thesis contains no material, which has been accepted for the award of any other degree or diploma in any tertiary institution. To the best of my knowledge and belief, the thesis contains no material previously published or written by another person, except when due reference is made in the text.

Statement of Authority of Access

This thesis may be made available for loan and limited copying in accordance with the *Copyright Act 1968*

Chainarong Klimanee

Abstract

Today, most biometrics research groups are tackling the challenging problem of an automatic fingerprint identification system (AFIS) using large databases. Since AFIS dedicates most of its processing time to searching for the best-matched fingerprint, searching over the entire fingerprint database is rather inefficient. It is proposed that the database be divided into sub-databases, each containing only fingerprints of the same pattern or class. Fingerprint classification is then an important first step in directing the search only to the appropriate sub-database, thus reducing the extent of searching of the large database.

The main objective of this thesis is to propose a classification technique to reliably classify a fingerprint into one of six well-known classes: *plain arch, tented arch, right loop, left loop, whorl* and *twin loop*. The fingerprint classification technique proposed in this thesis has achieved good results owing to the improvement in a number of processing steps the author has proposed for the *enhancement* of fingerprints, the determination of *singular points* and their associated *principal axes*, and the rule-based classification algorithm.

The directional bandpass *Gabor filter-bank* approach is one of the most effective and mathematically elegant techniques to date for fingerprint image enhancement. The filter output, however, is very sensitive to the ridge orientation and frequency that the filter is tuned to, and also to the spatial parameters of the Gaussian envelope. Unfortunately, filtering of a fingerprint image with an adaptive two-dimensional Gabor filter bank is computationally expensive because ridge orientation and frequency vary significantly throughout the fingerprint. In this thesis we propose to use an array of 8x4 two-dimensional Gabor filters tuned to eight directions and four ridge frequencies. Filtered fingerprint images at any combination of local ridge orientation and frequency can be calculated using a 2-D interpolation algorithm. The proposed technique produces a better quality of image than current Gabor-based techniques. The results are compared using a *goodness index* measure of the reliability of the automatic minutiae detection.

The accuracy of the location of singular points on a fingerprint is important for minutiae matching alignment and is also essential for the *Poincaré index* to correctly determine the type of singular points. In this thesis, we present a novel yet simple and accurate technique for the automatic determination of singular points. The technique offers a double-resolution estimation of the ridge orientation on a 4x4 pixels *quincunx* grid and quantises ridge orientations into six codes called *ridge flow codes. Singular regions* are defined as where all six codes exist. A singular point within a singular region can then be quickly determined by locating the point where the variance of local ridge orientation is at its maximum. The Poincaré indices of these singular points are used to determine their type: *ordinary, delta, core* or *double-core.* The distribution and type of the singular points, together with their associated principal axes, are then used to classify a fingerprint into one of six well-known classes or patterns.

This thesis proposes a rule-based algorithm for classifying fingerprints into one of six well-known classes. The rules are formed using the relative locations and types of singular points and the relative directions of their associated principal axes. The reliable and fast classification algorithm is made possible by a simple but effective combination of ridge flow-code technique and orientation variance calculation in the determination of singular points and principal axes. The Poincaré indices of these singular points are used to determine their type: ordinary, delta, core or double-core. For a test sample of some 150 fingerprints, the correct classification rate of the proposed algorithm was found to be better than 90%.

Acknowledgments

Firstly, I would like to express my deepest gratitude to my supervisor, Professor Thong Nguyen, for his valuable guidance, encouragement, support and consultation during my research. I greatly appreciate his enthusiasm in developing my research skills and advising on my thesis work.

Secondly, I would like to thank all staff and research colleagues in the School of Engineering in Hobart for helping me to complete my degree.

Table of Contents

List of Figures

List of Tables

Chapter One: Introduction

1.1 History of Fingerprints [1]

Fingerprints were impressed on clay tablets for conducting business in ancient Babylon. Thumbprints were also discovered on clay seals in China many hundreds of years ago.

Marcello Malpighi, a professor of anatomy at the University of Bologna, mentioned the spirals and loops in fingerprints in one of his notes in 1686 but said nothing about using the patterns for classification or identification of individuals. Nine fingerprint patterns were published in a thesis in 1823 by Professor John Evangelis Purkinji, a professor of anatomy at the University of Breslau. However, he also said nothing about using the patterns for classification or identification of individuals. In July 1858, fingerprints were used for the first time on native contacts by Sir William Herschel, Chief Magistrate of the Hooghly district in Jungipoo, India. Herschel was the first person who was convinced that fingerprints were unique to individuals, as well as being permanent throughout an individual's life. In more recent history, Dr. Henry Faulds, a British Surgeon-Superintendent of Tsukiji Hospital in Tokyo, Japan, was interested in 'skin-furrows' made by finger marks on samples of 'prehistoric' pottery. Consequently, Faulds presented innovative methods for identification and classification of fingerprints to the public, including to Charles Darwin in 1880. In the same year, the fingerprint and the use of printing ink for its acquisition was discussed as a means of personal identification in an article published in the Scientific Journal '*Nature*'.

Fingerprint identification was studied in 1880 by Sir Francis Galton, a British anthropologist and Charles Darwin's cousin. His book, entitled *Fingerprints*, was published in 1892. The book established the individuality and permanency of fingerprints. Furthermore, the first fingerprint classification system was reported in this book. Heredity and racial background were the primary interest of Galton for the purpose of identification of a person. Galton held a firm belief that fingerprints offered no firm clue to an individual's intelligence or genetic history, and as Herschel

and Faulds already suspected, that fingerprints do not change throughout an individual's life, and that no two persons have exactly the same fingerprint. Even the fingerprints of identical twins are different. By Galton's calculation, the probability of two individual fingerprints being the same was 1 in 64 billion. Galton used minutiae as the characteristics of fingerprint for identification. Nowadays, minutiae are still used and are often referred to as Galton's details. In 1892, the first criminal identification for proving the identity of a murderer was made by Juan Vucetich, an Argentinean police officer, using the bloody print left on a door post.

In England and Wales in 1901, Sir Edward Richard Henry began to use Galton's observation for fingerprint classification. The Henry Classification System is still in use today in all English speaking countries. In 1905, the U.S. Army began the use of fingerprints. During the next 25 years, U.S. law enforcement agencies were increasing the use of fingerprints for personal identification. One of the agency's officers, Edmond Locard, also proposed a specification for fingerprint identification. The number of minutiae points (Galton's Details) proposed for matching between two fingerprints accepted as positive identification was 12. However, some countries had set their own standard for the number of matching points, but not in the U.S.

By 1946, the USA's FBI kept some 100 million fingerprint cards in manually maintained files, and by 1971 the number had increased up to 200 million cards. Due to the introduction of the Automatic Fingerprint Identification System (AFIS), the files were separated into computerised criminal files and manually maintained civil files. Many of the manual files were duplicated for distributing to a neighbourhood of 25 to 30 million criminals and an unknown number of individuals in the civil files. Today, the FBI uses fingerprints as the primary method of identifying criminals. On the other hand, the technological advance of DNA testing has been developing to make identification more reliable but is not yet automatic. As a result, AFIS is still very popular in assisting the FBI in solving criminal cases.

1.2 Use of Biometrics [2,29]

Fingerprints are the most commonly used biometric features in various applications. Each type of biometrics has its own weaknesses and strengths, and its suitability

therefore depends on the application. A single biometric feature is usually insufficient for identifying a person. Therefore, a reliable biometric identification system is expected to incorporate many biometric characteristics that appropriately match the application. The most common biometric features are described in the following:

- **DNA**: DNA is abbreviated from Deoxyribo Nucleic Acid. The DNA is a one-dimensional feature of the ultimate unique code for one's individuality. Identical twins have the same DNA patterns. This fact is currently applied in the context of forensic applications for person identification [30]. The three recognised limitations of this biometric method are:
  (i) *contamination and sensitivity*: The piece of DNA from an innocent subject can be easily stolen and abused for criminal purposes,
  (ii) *automatic real time recognition issues*: DNA matching requires the expertise of DNA and the chemical methods are not set up for on-line non-invasive recognition, and
  (iii) *privacy issues:* Information about DNA is sensitive to a person having certain diseases. An example of DNA is shown in Figure 1.1.

Figure 1.1: An example of DNA characteristic [2].

- **Face:** Humans make a personal recognition by looking at the face of a person. The biometric characteristics of a human face are used in face recognition systems. It is a non-intrusive method. Applications of facial recognition [31] range from static and controlled 'mug-shot' verification to dynamic and controlled face identification in a cluttered background (e.g. airport). The most popular approaches to face recognition are based on either (i) the location and shape of facial attributes such as the eyes, eyebrows, nose, lips, and chin and their spatial

relationships, or (ii) the analytical model by which the face is represented as a weighted combination of a number of canonical elements. A face recognition system should automatically (i) determine that a face is present in the facial scanner, (ii) locate the face, if it has one, and (iii) recognise the face from any poses from a database. An example of a face is shown in Figure 1.2.

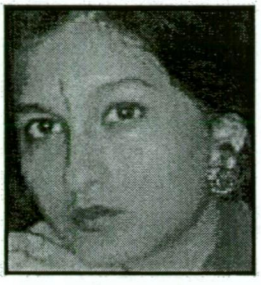


Figure 1.2: An example of a face [2].

- **Fingerprint:** This biometric feature has been well proven as a reliable means for personal identification. Fingerprint matching accuracy is very high. A fingerprint is formed from a pattern of ridges and valleys on the skin of a fingertip. Fingerprints are formed during the first six or seven months of foetal development. Today, low cost systems embedding fingerprint-based biometrics (e.g. laptop computers) have become affordable in a large number of applications. The accuracy of small-to-medium scale fingerprint recognition and identification systems involving a few hundred users are sufficient for many applications. Additional information about multiple fingerprints of a person is required in order to allow for large-scale recognition systems involving million of identities. A large amount of required computational resources is the main problem for current fingerprint recognition systems, especially when operating in the identification mode. Finally, the reliability of automatic identification systems is affected by fingerprints of a small fraction of the population because of genetic factors, aging, or environmental or occupational reasons (e.g. manual workers may have a large number of cuts and bruises on their fingers and these may keep changing). An example of a finger is shown in Figure 1.3.

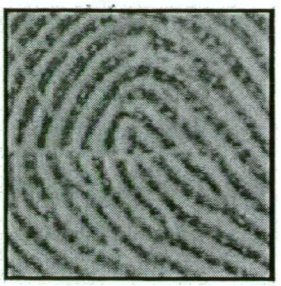


Figure 1.3: An example of a finger with cuts [2].

- **Hand and Finger Geometry:** A human hand is characterised by its shape, size of palm, and length and width of fingers. These features are used in hand geometry recognition systems [32]. Hand geometry-based verification systems have been commercialised and installed in hundreds of locations around the world. The advantage of the technique used for verifying hands is that it is simple, relatively easy to use, and inexpensive. This simple technique does not require details on the skin of the hand, and consequently the verification accuracy of the technique does not appear to be affected by dryness and other physical damage to the skin. The geometry of the hand is very distinctive because of its diverse variation. Furthermore, hand geometry, e.g. size of palm or length and width of fingers, may change during the growth period of children. In addition, acquiring hand geometry with rings on fingers, arthritis and an unnaturally posing hand, can confuse hand geometry-based recognition systems. Hand scanners are usually large because of the large size of hands. Some are smaller because they require only a few fingers (typically, index and middle) instead of an entire hand for verifying. An example of hand and finger geometry is shown in Figure 1.4.

Figure 1.4: An example of hand and finger geometry [2].

- **Iris:** The iris is an annular region of the eye segmented by the pupil and sclera (i.e. the white of the eye) on either side. In the foetal development period, the visual texture of the iris is formed and then is stabilised during the first two years of childhood. The distinctive complex iris texture is used as a reliable biometric feature for personal identification. The speed and accuracy of current iris-based recognition systems [33] today are very promising, and point to possible commercial large-scale identification systems. Iris and fingerprints are distinctive. Even the irises of identical twins are different. However, motivation for the use of iris texture is extremely difficult because it is easy for a person to use an artificial iris, and iris-based recognition systems are expensive. An example of an iris is shown in Figure 1.5.

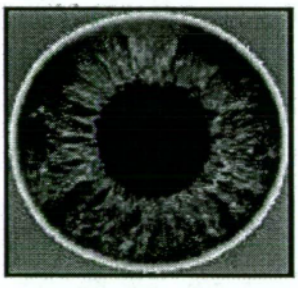


Figure 1.5: An example of an iris [2].

- **Signature:** The way people sign their names is known to be an individual characteristic. Although the contact with a writing instrument and the adjustment to an appropriate or comfortable way for signing do affect the way a person signs, most governments accept signatures as legal identification and verification of a person. Signatures are a behavioural biometric that change over a period of time. Furthermore, signatures are influenced by the physical and emotional conditions of signatories. The signature of some people is substantially variant of behaviour as even a successive impression of their signature can be significantly different. Further, a professional forger may be able to reproduce signatures that fool the system. However, the signature has been challenged in [34,35]. An example of a signature is shown in Figure 1.6.

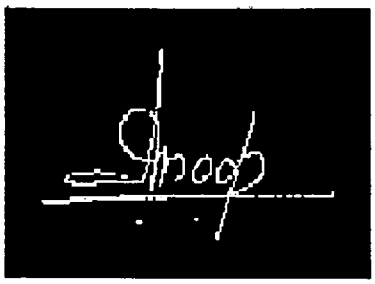


Figure 1.6: An example of a signature [2].

- **Voice:** Voice is a combination of physiological and behavioural biometrics. The features of an individual's voice are used for synthesis of sound as shape and size of the appendages (e.g. vocal tracts, mouth, nasal cavities, and lips). While speech is invariant for a person of given physiological characteristics, part of the speech of a person changes in behavioural characteristics due to age, medical conditions (such as the common cold), emotion, etc. In large-scale identification, voice may not be appropriate and not very distinctive. A voice recognition system [36,37] that is text-dependent is based on the utterance of a fixed predetermined phrase. A voice recognition system that is text-independent recognises a speaker regardless of what he/she says. The text-independent system is more difficult for designing algorithms than text-dependent systems but gives more protection against fraud. Background noise is a sensitive factor in degrading efficiency of voice recognition and this is a disadvantage for voice-based recognition. Many sophisticated telephone systems incorporate speaker recognition facilities because they are most appropriate and the most logical environments for phone-based applications. The voice signal, however, is often so degraded by the telephone set and by the communication channel that high accuracy for sophisticated applications cannot be guaranteed. An example of a voice signal is shown in Figure 1.7.

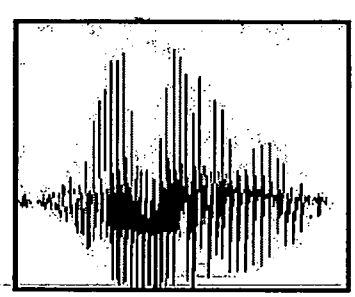


Figure 1.7: An example of a voice [2].

The requirements of use of biometric techniques depend on the purpose of application. Most operational environments cannot be manipulated by a single biometric technique. Consequently, the biometric systems used are mostly designed incorporating more than one technique in order to improve efficiency of the systems.

1.3 Advantages and Disadvantages of Fingerprint Identification/Verification

Advantages

In general, fingerprints have unique physical or behavioural characteristics that cannot be traded, shared, lost or stolen. Fingerprint recognition systems today can achieve sufficient accuracy for both verification and identification purposes. They are also an affordable and compact technology and are becoming more and more popular as a consumer product.

Disadvantages

Not everyone has fingerprints that can be recognised. The sensor is also not able to capture acceptable quality fingerprint images for people with very wet or very dry skin. In addition, the sensor needs to be maintained properly in order to achieve consistent performance.

1.4 Need for Fingerprint Classification

The ridge patterns information of fingerprints has been classified according to their global ridge patterns using the Henry System [1]. The well-known Henry System is over 100 years old. The main classes of ridge patterns are: *loop* (left loop and right loop), *arch* (plain arch and tented arch) and *whorl* (and twin loop). In a large fingerprint identification system, the large template database can be divided into sub databases using the ridge pattern classification to improve speed in identification mode. For example in Figure 1.8, if an acquired fingerprint is classified as plain arch, then the identification process searches for the matched fingerprint among templates only in the plain arch sub-database. This is very important when manipulating huge databases (like the FBI's) containing millions of fingerprints.

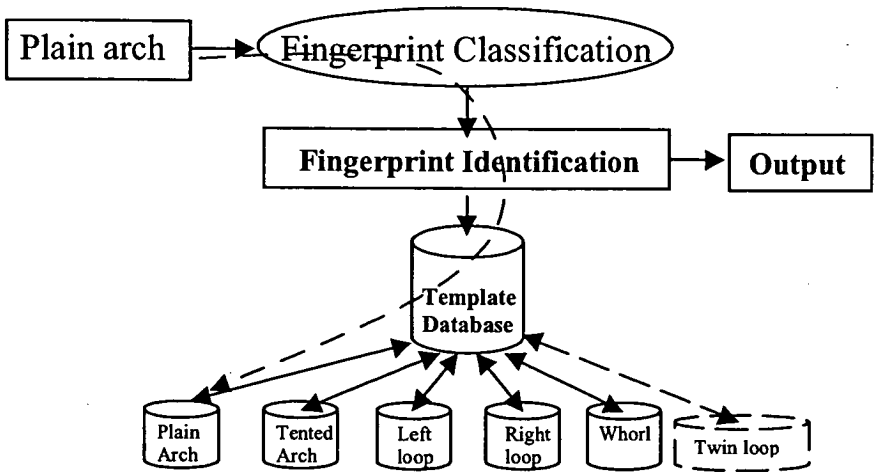


Figure 1.8: Example of a plain arch being first classified then identified by searching the sub database of plain arch only.

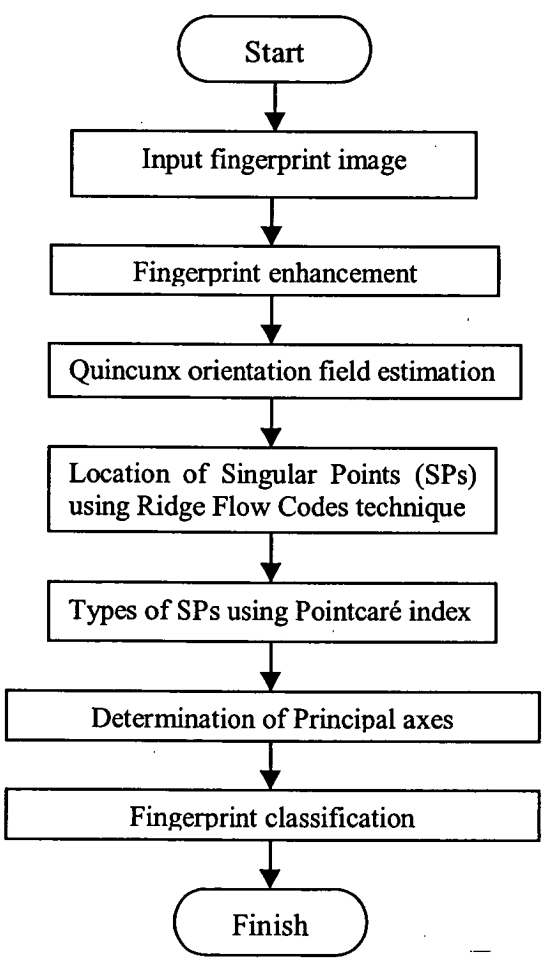


Figure 1.9: Overview of the fingerprint classification technique proposed in this thesis

1.5 Scope of the Thesis and the Main Contributions to the Field

1.5.1 Scope of the Thesis and Brief Description of the Main Processing Steps

The aim of the research work described in this thesis is to propose an accurate automatic fingerprint classification system. The system has the capability to classify a fingerprint into one of six classes: *plain arch*, *tented arch*, *left loop*, *right loop*, *twin loop* and *whorl*. A block diagram of an overview of the fingerprint classification system proposed in this thesis is shown in Figure 1.9. The technique assumes that the input fingerprint has not suffered from serious physical corruption or damage or missing parts. Techniques to handle these types of damaged fingerprints are beyond the scope of the thesis due to the time limit of a Masters thesis.

Fingerprint Enhancement: Fingerprint enhancement is applied for improving the clarity of the ridge structure of the input fingerprints. Apart from the standard pre-processing steps such as normalisation and histogram equalisation, filtering the input fingerprint with a Gabor filter has been very popular among researchers. Current techniques, e.g. [3,23,25] however, employ 2-D separable Gabor filter tuned to a fixed average ridge frequency. It is well known that on the same fingerprint the local ridge frequency can vary greatly. The new approach in this thesis [4] employs both orientation and frequency 2-D interpolation to obtain a filtered image with appropriate local parameters. Experimental results of our new approach have shown a significant improvement in clarity while retaining the original ridge structure.

Quincunx Orientation Field Estimation: Ridge orientation estimation is perhaps the most critical step in fingerprint processing. Its resolution and hence accuracy will affect the determination of other parameters of the fingerprint. Current techniques [3] mostly use a block resolution of no better than 8x8 pixels because of the validity of the statistical estimation and the computing time. Our technique achieves a doubled linear resolution for the estimation of ridge orientations, i.e. quadrupled number of orientation samples. This is achieved by an innovative sampling technique on a quincunx grid; thus requiring only doubled computing time.

Determination of Singular Points (SPs) using Ridge Flow Codes Technique: The location of an SP is determined using the newly proposed *ridge flow codes* technique. The codes are obtained by a simple quantisation of ridge, and the mapping of these codes over the fingerprint provides a compact and informative ridge flow pattern. The location of an SP is defined as the point of maximum orientation variance in a singular region, and therefore is the meeting point of as many flow codes as possible. SPs are classified into three types: *ordinary*, *core* and *delta* using the Poincaré index. In current techniques [5,28], the Poincaré index is used for concurrently determining both the location and type of an SP. Our new technique proposes to save time by first quickly locating the singular regions from the ridge flow map, then calculating the Pointcaré index only to pixels in those regions to finally determine the location and type of SPs.

Determination of Principal Axes: This is one of the main contributions to the field from this thesis. The types and locations of SPs are insufficient for fingerprint classification purposes. Consequently, we introduce the new concept of principal axes associated with a singular point [6]. Principal axes, representing the dominant directions of ridge flow in the immediate vicinity of a core point, serve as valuable quantitative ridge flow features for creating rules for fingerprint classification.

Fingerprint Classification: The characteristics SPs (type and location) together with the characteristics of their associated principal axes (number and relative orientation) are employed in establishing rules for fingerprint classification. The two distinct patterns of whorl and twin loops, both containing two cores and two deltas, continue to be a challenge to researchers to automatically distinguish them. The added challenge is that quite often the deltas, being usually located near the lower edge of the finger, are missing from the acquired fingerprint. Consequently, whorls and twin loops are incomplete and are rejected by our classification system. The proposed rules in this thesis for fingerprint classification has the ability to classify fingerprints into six classes. Current techniques combine whorl and twin loop into one single class to avoid the challenge mentioned above.

1.5.2 List of Published Papers

C. Klimanee and D.T. Nguyen, "On the Design of 2-D Gabor Filtering of Fingerprint Images," *Proceedings IEEE International Consumer Communications and Networking Conference CCNC'04*, Caesar's Palace, Las Vegas, Nevada USA, 6-9 January 2004.

C. Klimanee and D.T. Nguyen, "Classification of Fingerprints using Singular Points and Principal Axes," *Submitted to IEEE International Conference in Image Processing ICIP'04,* Singapore, 24-27 October 2004.

C. Klimanee, A.V. Trinh, and D.T. Nguyen, "Accurate Determination of Singular Points and Principal Axes on a Fingerprint," *Submitted to IEEE Region 10 International Conference TENCON2004,* Chiang Mai, Thailand, 21-24 November 2004.

1.6 Structure of the Thesis

The structure of the thesis is briefly outlined as follows:

Chapter 2 describes the pre-processing steps such as histogram stretching, histogram equalisation using a sigmoid function, and normalisation. These techniques are used for removing noise and improving clarity of images. However, the techniques can improve the clarity of the ridge and valley structure.

Chapter 3 describes the parameters estimations necessary for the enhancement of fingerprints using a Gabor filter. This includes the estimation of the orientation field, the estimation of local ridge frequency, and the estimation of frequency range.

Chapter 4 describes the directional Gabor filter in spatial and frequency domain. In this chapter, we define and explain the meaning of various parameters of the Gabor filter for filtering fingerprints, and the roles they play in the response characteristics of the filter. The main contribution of this chapter is the design of a separable 2-D directional Gabor filter from its 1-D counterpart. A 2-D interpolation algorithm is

proposed to adaptively tune the filter to the correct local ridge orientation and frequency of ridge in order to obtain a better output filtered image.

Chapter 5 describes the segmentation of the fingerprint into foreground and background areas before the minutiae extraction process.

Chapter 6 studies the operations of the mathematical morphology necessary for minutiae extraction such as opening, closing and hit-and-miss operation. These operations are used for thinning binary images before minutiae detection is possible. The accuracy of the minutiae extraction is used for assessing the performance of the proposed enhancement technique using a *goodness index*.

Chapter 7 proposes a new Singular Point detection technique making use of the efficiency of our high-resolution technique for orientation estimation. This chapter describes the detection of SPs using ridge flow codes and the Pointcaré index. The sensitivity of the Pointcaré index to the location of the summing contour is carefully studied and presented. Furthermore, the new approach for determining principal axes using the histogram technique with an orientation mask is also covered in this chapter.

Chapter 8 proposes a rule-based algorithm for classifying a fingerprint into one of six well-known classes: plain arch, tented arch, right loop, left loop, whorl and twin loop. The rules are derived from the characteristics of SPs and their associated principal axes.

Finally, Chapter 9 concludes the thesis with a direction for future research and improvement.

Chapter Two: Pre-processing Techniques

2.1 Introduction

This chapter presents the pre-processing techniques used in this thesis for enhancing the contrast of fingerprint images, but not the clarity of the ridge and valley structure. The three techniques chosen are contrast stretching, contrast enhancement using sigmoid function, and the normalisation technique. The first operation, the contrast stretching technique is applied to normalise the intensity of the image from the original range to the full range of greyscale (0-255). The second operation, the contrast enhancement using a sigmoid function, is used to enhance the contrast of the image by emphasising dark and light regions in the histogram. Finally, the normalisation technique is applied to normalise the image to a given desired mean and variance.

This chapter is organised as follows: Section 2 explains the contrast stretching technique. Section 3 describes the contrast enhancement using sigmoid function. Section 4 explains the normalisation technique.

2.2 Contrast stretching [7]

This technique is used to improve an image by stretching the range of intensity values from the original range to a new range. It is a simple linear mapping of the input to the output. For example, a fingerprint image in greyscale is normalised from the original range (lower = 0 and upper = 211) to a new range (e.g. 0 – 255 for a standard 8-bit greyscale picture) by

$$s = (r - c)\left(\frac{b-a}{d-c}\right) + a \qquad (2.1)$$

where

r: Input pixel value of the original image

s: Output pixel value of the new image

a: Lower limit of the new range

b: Upper limit of the new range

c: Lower limit of the original range

d: Upper limit of the original range

An experimental result is shown in Figure 2.1 in which the histogram of the original image in (c) is stretched to give the histogram in (d).

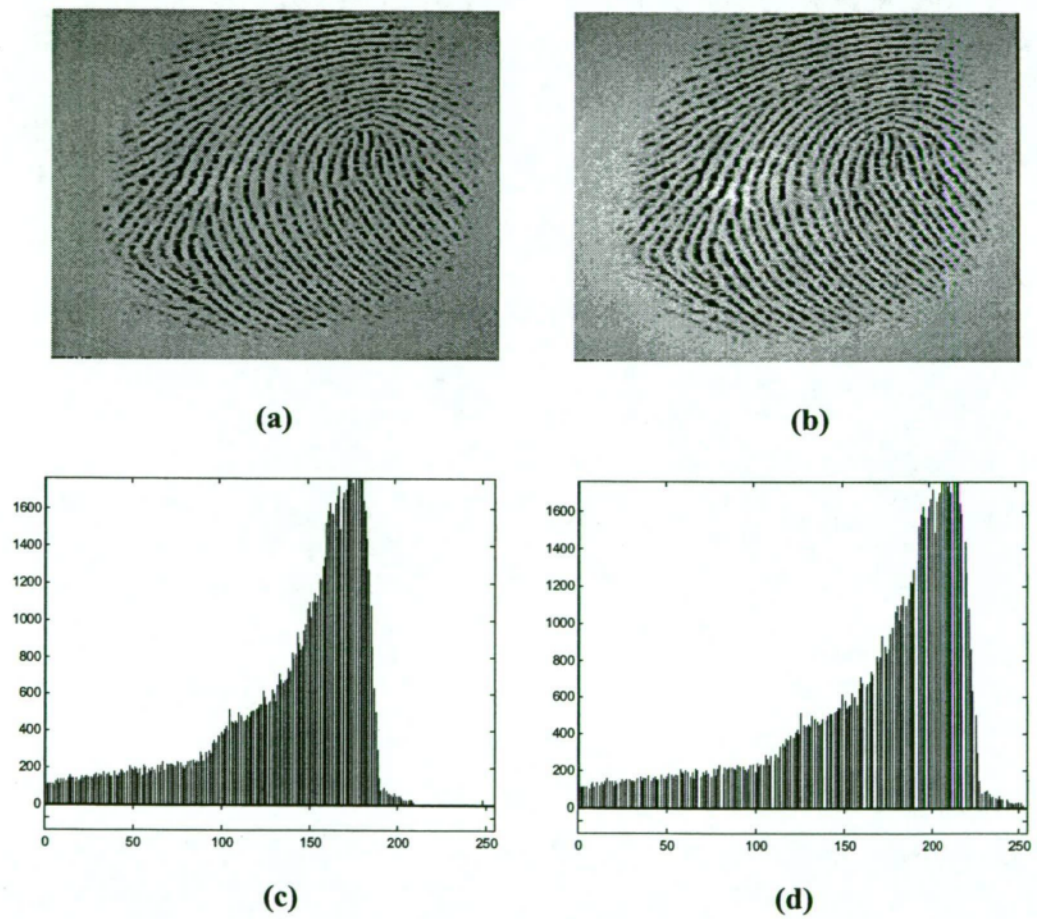


Figure 2.1: (a) Original image, (b) Image after contrast stretching, (c) Histogram of the original image in (a), and (d) Histogram of the enhanced image in (b).

2.3 Contrast enhancement using sigmoid function

The function in (2.2) is used to give the sigmoid shape in Figure 2.2 in order to enhance the contrast of a greyscale image. Intensities above the mid-value of the histogram are lightened while those below are darkened.

$$Out = MeanI(1 - cos(\frac{In}{255} \times \pi)) \qquad (2.2)$$

where

Out: Pixel value of the enhanced output image

Mean: Mean value of pixels in the original image

In: Pixel value of the original input image

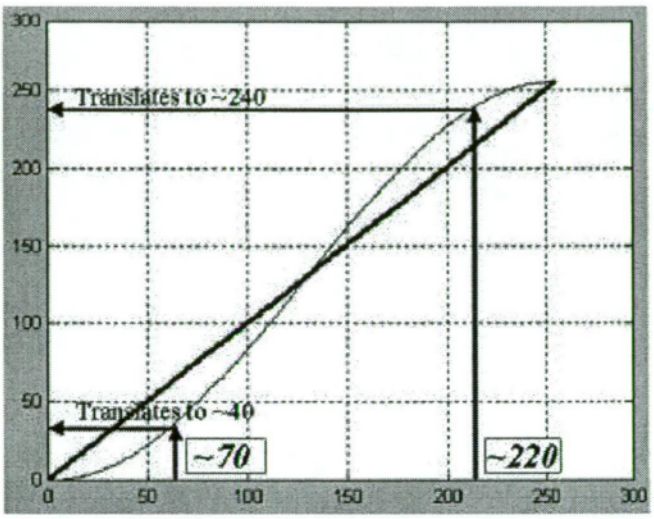
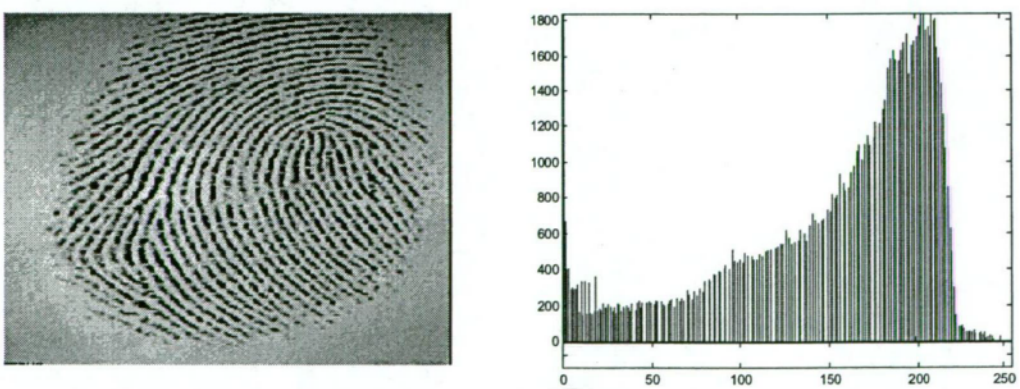


Figure 2.2: Sigmoid shape with mean equal to 128 (thin line) and position of maximum sigmoid shape in the histogram.

The result shown in Figure 2.3, after applying the sigmoid-shaped mapping function to the original image, gives an obvious contrast improvement. The histogram in Figure 2.3 (b) has been enhanced in the dark and light regions compared to the histogram of the original image shown in Figure 2.1 (c).



(a) **(b)**

Figure 2.3: (a) Image after contrast enhancement and (b) Histogram of the image in (a).

2.4 Normalisation [8,21,27]

Most input fingerprint images contain a varying greyscale across the fingerprint that can be improved using a normalisation technique. This technique can improve the clarity of the image but it cannot enhance the details of the ridge and valley structures.

Let $I(i,j)$ denote the grey level of a pixel at coordinates (i,j) in the image; M_i and V_i denote the mean and variance, respectively, calculated over the entire original image; M_d and V_d denote the desired mean and variance, respectively; and $N(i,j)$ denotes the normalised grey level of the pixel at (i,j). The normalisation of the image's grey levels is then given in Equation (2.3).

$$N(i,j) = \begin{cases} M_d + \sqrt{\dfrac{V_d (I(i,j) - M_i)^2}{V_i}} & \text{if } I(i,j) > M_i \\ M_d - \sqrt{\dfrac{V_d (I(i,j) - M_i)^2}{V_i}} & \text{otherwise} \end{cases} \qquad (2.3)$$

Figure 2.4 shows a result of normalisation.

(a) **(b)**

Figure 2.4: (a) Original image, and (b) Normalised image

2.5 Conclusion

The three techniques mentioned above have been used in this thesis for the pre-processing of fingerprint images. The pre-processing starts with contrast stretching, followed by an enhancement using a sigmoid function, then finishes with a

normalisation of the mean and variance of the image. The pre-processing gives a clear contrast improvement to the image as shown in Figure 2.5.

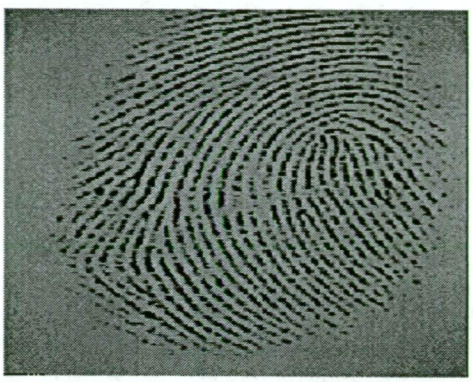


Figure 2.5: Contrast-improved fingerprint image using the three pre-processing techniques.

Chapter Three: Parameter Estimation Techniques

3.1 Introduction

This chapter presents the techniques for estimating parameters of the enhancement technique in Chapter 4. Gabor filter is both frequency-selective and orientation-selective and can be tuned to corresponding frequency and orientation of local ridges in order to improve clarity of ridge structure and remove undesired noise in fingerprint image. Consequently, the Gabor filter is used for enhancing a fingerprint. Therefore, those parameters estimations will be mentioned with the technique involved as frequency range estimation in this chapter.

This chapter is organised as follows: Parts 2, 3 and 4 explain the local ridge orientation, local ridge frequency and frequency range estimations, respectively.

3.2 Local Ridge Orientation Estimation

The ridge orientation field is a map of local block orientations. A block orientation is defined as the most dominant of ridge orientations at all the pixels in the wxw (usually 8x8 or16x16) block. Let the estimated orientation field be $\theta(i,j)$ where (i,j) are the coordinates of the centre pixel of the block, and there is an implied ambiguity of 180^0 in the ridge direction. It is obvious that the dominant ridge orientation of a block corresponds approximately to the direction perpendicular to the dominant direction of the power spectrum of the same block. The latter is exactly the dominant direction of the ridge gradient ϕ. In this chapter we use the least mean square algorithm in [9] to estimate $\phi(i,j)$ then $\theta(i,j) = 90^0 - \phi(i,j)$. The result as given in [8,20,24] is

$$\phi(i,j) = \frac{1}{2}\tan^{-1}\left[\frac{V_y(i,j)}{V_x(i,j)}\right] \tag{3.1}$$

in which the vertical and horizontal contributions are respectively

$$V_y(i,j) = \sum_{x=i-w/2}^{i+w/2} \sum_{y=j-w/2}^{j+w/2} 2G_x(x,y)G_y(x,y) \tag{3.2a}$$

and

$$V_x(i,j) = \sum_{x=i-w/2}^{i+w/2} \sum_{y=j-w/2}^{j+w/2} \{G_x^2(x,y) - G_y^2(x,y)\} \quad (3.2b)$$

where the horizontal gradient $G_x(x,y)$ and the vertical gradient $G_y(x,y)$ at each pixel (x,y) may be calculated using a simple 3x3 Sobel edge mask or a more sophisticated gradient operator.

Due to the presence of noise and distortion, the estimated orientation field is not smooth. In order to carry out lowpass filtering (at the block level) on the orientation field, the filtering has to be performed on the components $V_x(i,j)$ and $V_y(i,j)$ separately. 2-D Lowpass filtering using a simple 5 block x 5 block kernel W(m,n) is used to give smoothed $V_x'(i,j)$ and $V_y'(i,j)$. The final smoothed local ridge gradient becomes

$$\phi(i,j) = \tfrac{1}{2}\tan^{-1}\left(\frac{V_y'(i,j)}{V_x'(i,j)}\right) \quad (3.3)$$

Figure 3.1 shows the ridge orientation field θ(i,j) of a scratched fingerprint that is used as a standard sample for the international fingerprint verification competition FVC2002 [10].

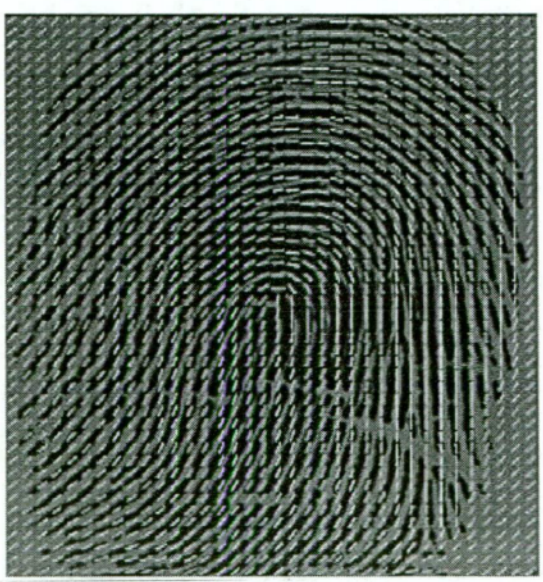
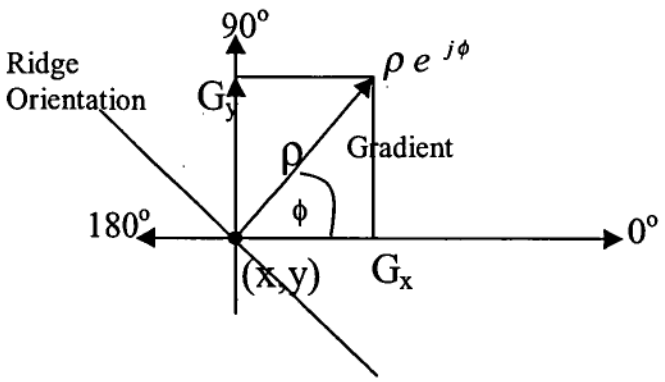


Figure 3.1: Ridge orientation field superimposed on fingerprint image.

The proof of equation (3.1) is presented in the following:



Let the local ridge gradient be

$$\Delta I(x,y) = \rho e^{j\phi} = G_x + jG_y$$

where

$$G_x = \rho\cos\phi \quad \text{and} \quad G_y = \rho\sin\phi\,.$$

Then the square of gradient is

$$(\Delta I)^2 = \rho^2 e^{j2\phi} = \rho^2[\cos 2\phi + j\sin 2\phi]$$

$$= \frac{\rho^2\sin 2\phi}{\rho^2\cos 2\phi} = \left[\frac{2\rho^2\sin\phi\cos}{\rho^2(\cos^2\phi - \sin^2\phi)}\right] = \left[\frac{2G_xG_y}{G_x^2 - G_y^2}\right]$$

$$Mean\ square\ of\ (\Delta I) = \left[\frac{\sum_{x=i-w/2}^{i+w/2}\sum_{y=j-w/2}^{j+w/2} 2G_xG_y}{\sum_{x=i-w/2}^{i+w/2}\sum_{y=j-w/2}^{j+w/2}(G_x^2 - G_y^2)}\right], \quad \text{let} = \left[\frac{V_y}{V_x}\right]$$

The most dominant gradient direction is therefore,

$$\phi = \frac{1}{2}\tan^{-1}\frac{V_y}{V_x}$$

or

$$\phi = \frac{1}{2}\tan^{-1}\frac{\sum_{x=i-w/2}^{i+w/2}\sum_{y=j-w/2}^{j+w/2} 2G_xG_y}{\sum_{x=i-w/2}^{i+w/2}\sum_{y=j-w/2}^{j+w/2}(G_x^2 - G_y^2)}$$

3.3 Local Ridge Frequency Estimation

To estimate the local ridge frequency we adopt the x-signature approach in [8]. Figure 3.3 illustrates the technique. For each 8x8 pixels block centred at (i,j), we define a rectangular lxw=32x8 pixels window, oriented at right-angles to the ridge direction θ(i,j) estimated from ϕ(i,j) in (3.3) above. Next we compute the x-signature {X(0), X(1), ...X(31)}, i.e. the image intensity variation along the window orientation, where each X(k) is the average value along the ridge and valley direction within the rectangular window, i.e.

$$X(k) = \frac{1}{w}\sum_{d=0}^{w-1} I_d(u,v) \tag{3.4}$$

where the co-ordinates

$$\begin{aligned} u &= i + (d - w/2)\cos\theta(i,j) + (k - l/2)\sin\theta(i,j), \\ v &= j + (d - w/2)\sin\theta(i,j) - (k - l/2)\cos\theta(i,j). \end{aligned} \tag{3.5}$$

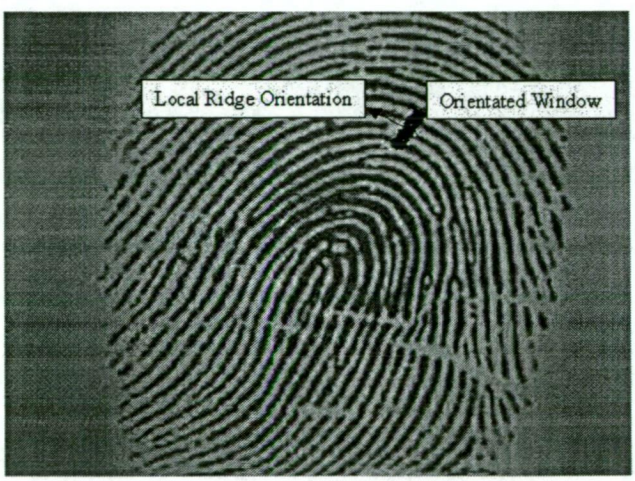


Figure 3.2: Oriented window superimposed on fingerprint image in the direction of the local ridge orientation.

As an example, consider the local ridge orientation of a block centred at coordinates (i=250, j=100). The block is highlighted by the black bar on the fingerprint image in Figure 3.2. The coordinates in the orientated window are calculated by (3.5).

If there are no minutiae and other singular points in the window, the x-signature forms a sinusoidal wave, and the local ridge frequency can be reliably estimated. For the FBI scanning standard of 500dpi, this frequency ranges from 3 to 25 pixels per pitch

between ridges. If the estimated inter-ridge pitch is outside this range, or if the x-signature does not form a well-defined sinusoidal wave, the estimated frequency of the block is rejected. If there are not too many of these bad blocks, their local frequencies can be interpolated from the frequencies of the neighbouring blocks.

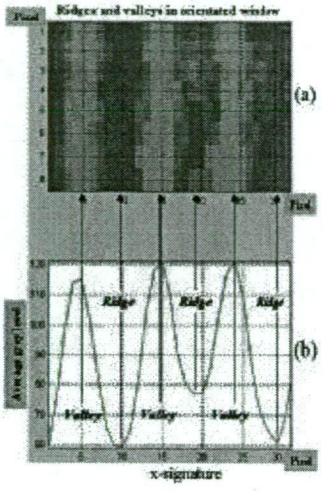


Figure 3.3: X-signature profile of local ridge-valley structure.

Ridges and valleys along the local orientated window are shown in Figure 3.3(a). Equation (3.4) estimates the x-signature shown in Figure 3.3(b). Ridge frequency f (u,v) can be estimated from the peak-to-peak pitch values of the sinusoidal wave of the x-signature. The processing steps of the estimation are as follows:

Step1: Estimate the local ridge orientation in the block centred at (i,j).

Step2: Define a rectangular window, oriented along the local ridge direction of the block centred at (i,j).

Step 3: Calculate the average intensity of ridge and valley in the rectangular window in order to obtain the x-signature. Figure 3.3(b) shows the x-signature of averaged intensity of ridge and valley in the rectangular window.

Step 4: Apply a low-pass filter to the x-signature to eliminate undesired noise. Figure 3.4 shows the x-signature before and after filtering.

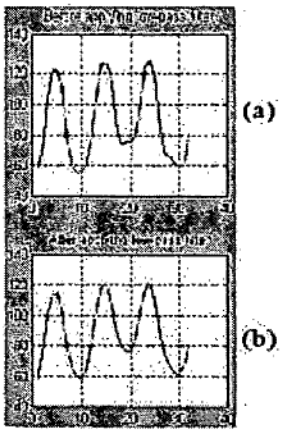


Figure 3.4: (a) X-signature before applying low-pass filter. (b) X-signature after applying low-pass filter.

Step 5: Encode the grey level of the x-signature in Figure 3.5(a) to bipolar code in Figure 3.5(b) as follows:

If the signal increases, encode it to +1.

If the signal stays unchanged, encode it to 0.

If the signal decreases, encode it to -1.

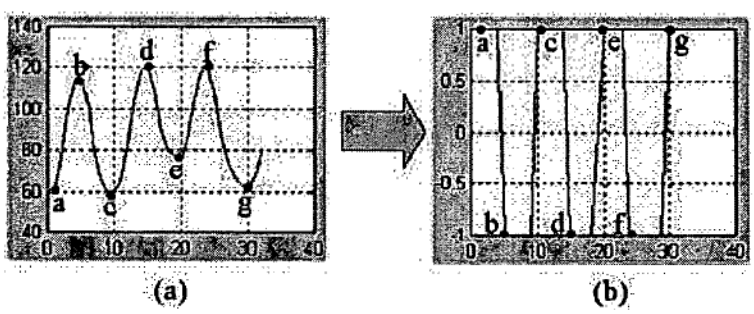


Figure 3.5: (a) X-signature before encoding, (b) Bipolar codes of x-signature.

Step 6: Extract position of the peaks (maxima and minima) from the bipolar code where the code changes.

Step 7: Calculate ridge frequency by averaging the number of pixels between peaks.

Since every sinusoid goes through two peaks – one maximum and one minimum - the local ridge period is the distance between two maxima or two minima.

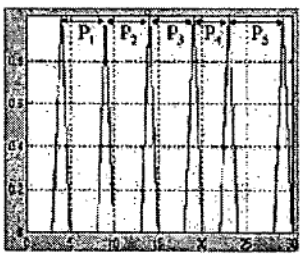


Figure 3.6: Peak positions in binary form.

Let P_i be the number of pixels between peak positions (maxima and minima), and N be the number of peak intervals. The local ridge frequency is calculated by

$$Ridge\_frequency = [\ \frac{2}{N}\sum_{i=1}^{N} P_i\ ]^{-1} \tag{3.6}$$

3.4 Frequency Range Estimation

Ridge frequencies in any particular fingerprint image usually do not cover the entire frequency range recommended by the FBI (from 3 to 25 pixels per pitch between adjacent ridges). At the same time, directional Gabor filtering of fingerprint images is very computationally expensive. Frequency range estimation is therefore an economical necessity so that the fingerprint is filtered only into those directions within the estimated range, i.e.

$$u_{oi} = \{\min(F_i),...,\max(F_i)\} \tag{3.7}$$

3.5 Conclusion

In this chapter, the two current techniques [8] have been presented. The orientations of ridge are accurately estimated using gradient technique with lowpass filter. Furthermore, the frequencies of ridge are also accurately estimated by applying estimated orientations to create an orientated window for forming x-signature that is in simple form used for estimating frequency of ridge. In order to reduce the number of computations, the estimation of frequency range is employed to increase speed of process.

Chapter Four: Directional Gabor Filter and Enhancement Techniques

4.1 Introduction

The main objective in fingerprint image enhancement in a minutiae-based automatic fingerprint recognition system is for reliable detection of minutiae. These delicate features, however, are extremely sensitive in both location and type (e.g. ridge ending or bifurcation) to spatial distortion and impairment of the ridges caused by inappropriate design of the enhancing filter's parameters. In most practical situations, acquired fingerprint images are very poor or even unusable. Good image enhancement techniques can recover some corrupted regions but other regions remain unrecoverable. Fingerprint image restoration is not the main topic of this chapter and we assume that the fingerprint has not suffered from serious physical distortion or corruption.

Directional 2-D Gabor filtering [3] has been one of the most significant and most mathematically elegant enhancement algorithms for fingerprint images to date. When the local frequency and local orientation of a pattern of parallel lines (e.g. the ridges on a fingerprint) are reasonably well defined, a directional bandpass filter tuned to the parallel frequency and perpendicular orientation can be used to remove noise and artefacts that interfere with the regular pattern. A 2-D directional Gabor filter is most suitable for this function because it is both frequency-selective and orientation-selective and has optimal joint resolution in both spatial and spatial-frequency domains. However, directional bandpass filtering of a fingerprint is computationally expensive because the ridge's spacing (known as pitch) and orientation vary significantly throughout the finger. The common practice for estimating the local ridge orientation (being perpendicular to the ridge gradient) and ridge frequency (being the inverse of pitch) is to partition the image into non-overlapping square blocks and to use some optimisation techniques to calculate the most dominant orientation and frequency of every block.

In [3], a directional Gabor filter bank tuned to eight evenly spaced directions at 22.5^0 apart – a 180^0 peacock-tail configuration, is used to give eight directional filtered images. All filters in the bank, however, are tuned to the same centre frequency u_0 being fixed at the average frequency over the entire image. The image at any other direction is obtained by using a simple linear interpolation. In a directional image, those ridges and valleys that have local orientation, being close to the tuned direction of the filter, are expected to appear strongest. Unfortunately, this is not always the case because in some areas ridges and valleys have a frequency significantly different from that of the filters. A rather computationally intensive 'voting algorithm' is proposed in [3] to select the strongest filtered response among the eight filtered images for every block in the fingerprint. The resulting image, quilted from the selected blocks, will hopefully give the optimally enhanced fingerprint image. The voting algorithm sometimes picks the wrong direction because the strongest intensity is due to its local frequency being close to the filter's frequency u_0. This results in a visible blocking effect due to ridge orientation mismatch with neighbouring blocks.

Perhaps the most serious drawback of having fixed centre frequency u_0 is that the filter's spatial parameters are consequently also fixed. If the Gaussian envelope is too narrow, the filter's passband will be broad enough to also detect the second harmonic of the tuned frequency, i.e. $2/u_0$, causing what is known as ridge splitting with unpredictable appearance and disappearance of minutiae. On the other hand, if the Gaussian envelope is too broad, it can merge or bridge neighbouring ridges and valleys. The merging of two ridges can result in a ridge ending becoming a ridge bifurcation, whilst the bridging of two valley endings can even result in the disappearance of the ridge bifurcation altogether.

In this chapter, we propose a more accurate and not necessarily more computationally expensive algorithm for computing the Gabor filtered image at any given orientation and frequency of the ridges and valleys in a local block. The voting algorithm in [3] is avoided in our algorithm.

The most successful fingerprint recognition techniques to date can be broadly classified as either being minutiae-based (most active at Michigan State University) or correlation-based (most active at the University of Twente). In minutiae-based

techniques, two sets of minutiae points are aligned and the total number of matched minutiae is used as the matching score [3]. These techniques require accurate detection of minutiae points which is a very challenging task, and complex dynamic matching techniques for matching two sets of minutiae points so that one is usually the 3-D geometric distortion of the other. Correlation-based techniques avoid these stringent requirements by matching the global pattern of ridges and valleys [12]. The global characteristics, however, are sensitive to non-linear 3-D distortions and noise present in the image, although many authors have concluded that minutiae-based techniques perform better than correlation-based techniques.

This chapter is organised as follows: Part 2 briefly presents the Gabor transform. Part 3 presents a definition of variables and functions that are used for the design of the Gabor filter. Part 4 explains the meaning of the units of the parameters that characterise fingerprint images. Part 5 presents the 1-D Gabor filter that is the basis for the generation of 2-D Gabor filters. Part 6 describes the 2-D Gabor filter. Part 7 derives a directional 2-D Gabor filter in the spatial domain by a coordinate transformation of the 2-D Gabor filter in Part 6. Part 8 shows how to estimate the parameters of the Gabor filter for filtering fingerprint using visual estimation. Part 9 illustrates the use of the Gabor filter for filtering fingerprints. Part 10 derives a 2-D directional Gabor filter in the spatial- frequency domain by taking the Fourier transform of its spatial counterpart. Part 11 studies the use of the Gabor filter in spatial-frequency domain and illustrates the steps in the processing of fingerprint images. Part 12 proposes the new technique as 2-D interpolation of Gabor filter for purposes of image enhancement. Part 13 illustrates the processing steps of the new enhancement technique using 2-D interpolation.

4.2 Gabor Transform

Dennis Gabor was born in Hungary in June 1900. He was a physicist and engineer, who gained British citizenship in 1946. The core of Gabor's analysis is to expand a signal in a series of time-shifted and frequency-modulated Gaussian functions. It is known as the Gabor expansion. Using the Gabor expansion, the behaviour of a signal can be analysed simultaneously in time and frequency domains. Gabor expansion applies Gaussian function for optimising the concentration of the signal's energy.

Since a Gaussian modulated sinusoid can be considered as a windowed sine wave, Gabor transform is often referred to as a short-time Fourier transform (STFT). It is a modified Fourier transform with a time-shifted Gaussian window that results in a time-dependent spectrum. The Gabor expansion and the STFT are therefore mutually equivalent.

4.3 Definition of Variables and Functions

x : spatial distance in horizontal direction (in pixels).

g(x): Gaussian function in x-direction

σ_x: standard deviation of the Gaussian function in x-direction (in pixels)

u: spatial frequency in horizontal direction (in cycles/pixel or cpp)

σ_u $= \frac{1}{2\pi\sigma_x}$, standard deviation of the Gaussian filter in the horizontal frequency (in cpp)

y : spatial distance in vertical direction (in pixels)

g(y): Gaussian function in y-direction

σ_y: standard deviation of the Gaussian function in y-direction (in pixels)

v: spatial frequency in vertical direction (in cycles/pixel or cpp)

σ_v: $= \frac{1}{2\pi\sigma_y}$, standard deviation of Gaussian filter in the vertical frequency (in cpp)

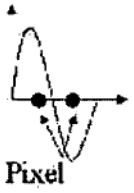
ϕ: direction/orientation of the Gabor filter (direction of maximum frequency response)

P_o: pitch of ridges (distance in pixels between 2 adjacent ridges)

u_o: $= \frac{1}{P_o}$, tuned frequency of the Gabor filter in the direction at right angles to the ridges (in cpp)

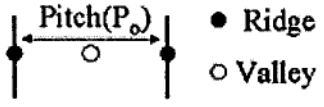
4.4 Units

cycles/pixel (cpp): number of cycles per pixel. The highest value of this unit in digital signal processing is two pixels per cycle or 0.5 cycle/pixel.



0.5 cycle/pixel

pixels/ridge (Pitch): A pitch is defined as the distance (in number of pixels) between two ridges (maxima) or between two valleys (minima).



4.5 One-Dimensional Gabor Filter in Spatial Domain

1-D Gabor filter is in effect a windowed sinusoid basis function of a short-time Fourier transform (STFT). Let $f(x,u_o)$ be the sinusoid basis function having frequency u_o, i.e.

$$f(x,u_o) = \cos(2\pi u_o x) \tag{4.1}$$

Let $g(x)$ be the Gaussian window function having standard deviation σ_x,

$$g(x,\sigma_x) = \frac{1}{\sigma_x\sqrt{2\pi}}\exp\left(-\frac{x^2}{2\sigma_x^2}\right) \tag{4.2}$$

1-D Gabor Filter is derived from $f(x)$ and $g(x)$ by multiplying or windowing $f(x)$ with $g(x)$. Therefore, 1-D Gabor filter in x-direction is

$$GlD_x(x;u_o,\sigma_x) = \frac{1}{\sigma_x\sqrt{2\pi}}\exp\left(-\frac{x^2}{2\sigma_x^2}\right)\cos(2\pi u_o x) \tag{4.3}$$

where $u_o = \frac{1}{P_o}$ (4.4)

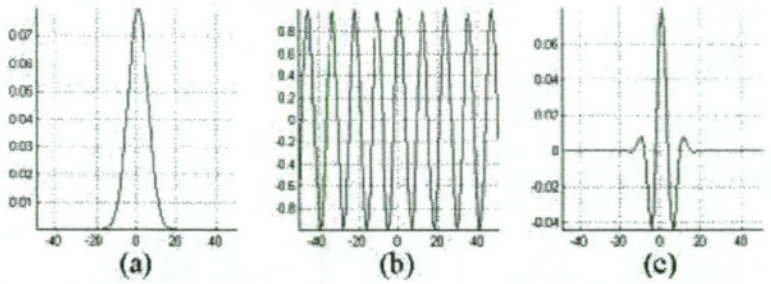


Figure 4.1: 1-D Gabor filter in spatial domain: (a) Gaussian envelope, (b) Sinusoid function, and (c) 1-D Gabor filter in x-direction.

Figure 4.1 illustrates the components of a 1-D Gabor filter in which (a), (b) and (c) are generated by using $g(x,\sigma_x)$, $f(x,u_o)$ and $G1D_x(x;u_o)$ respectively with σ_x =5 pixels and u_o=1/10 cpp, i.e 10 pixels between two adjacent ridges.

4.6 Two-Dimensional Gabor Filter in Spatial Domain

A separable 2-D Gabor filter is derived from its two 1-D components $G1D_x(x;u_o)$ and $G1D_y(y;u_o)$. However since we are interested in the tuned frequency in the x-direction only, the 2-D Gabor filter in x-direction is $G2D(x,y;u_o)$. Figure 4.2 illustrates the spatial response of the 2-D Gabor filter with σ_x = 5 pixels, σ_y = 5 pixels and u_o=1/10 cpp.

$$G2D(x,y,u_o,\sigma_x,\sigma_y) = G1D_x(x,u_o,\sigma_x) \,.\, G1D_y(y,\sigma_y) \qquad (4.5)$$

$$G2D(x,y,u_o,\sigma_x,\sigma_y) = \frac{1}{2\pi\sigma_x\sigma_y}\exp\left(-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2}+\frac{y^2}{\sigma_y^2}\right)\right)\cos(2\pi u_o x) \qquad (4.6)$$

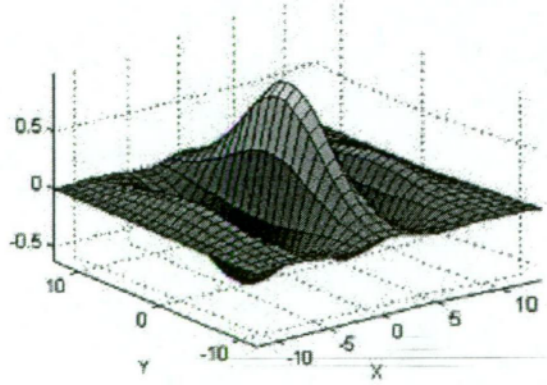


Figure 4.2: Two-dimensional Gabor filter in spatial domain modulated in x-direction

4.7 Directional Gabor Filter in Spatial Domain

The filter G2D(x,y,u_o,σ_x,σ_y) can be rotated to another direction ϕ but keeping its spatial response the same in the new coordinates (x',y'). Therefore

$$G2D(x',y',u_o,\sigma_x,\sigma_y,\phi) = \frac{1}{2\pi\sigma_x\sigma_y}\exp\left(-\frac{1}{2}\left(\frac{x'^2}{\sigma_{x'}^2}+\frac{y'^2}{\sigma_{y'}^2}\right)\right)\cos(2\pi u_o x') \quad (4.7)$$

where $x' = x\cos\phi + y\sin\phi$

$y' = y\cos\phi - x\sin\phi$

Note that we always plot the response of the filter on the original coordinates (x,y), i.e. we calculate (x',y') for given (x,y), and then calculate the spatial response G2D (x',y';u_o,ϕ) from the equation above. Figure 4.2 corresponds to the directional Gabor filter with ϕ=0, i.e. maximum response along x-direction suitable for the detection or filtering of vertical ridges. Figure 4.3 illustrates the response of the 2-D directional Gabor filter in various directions.

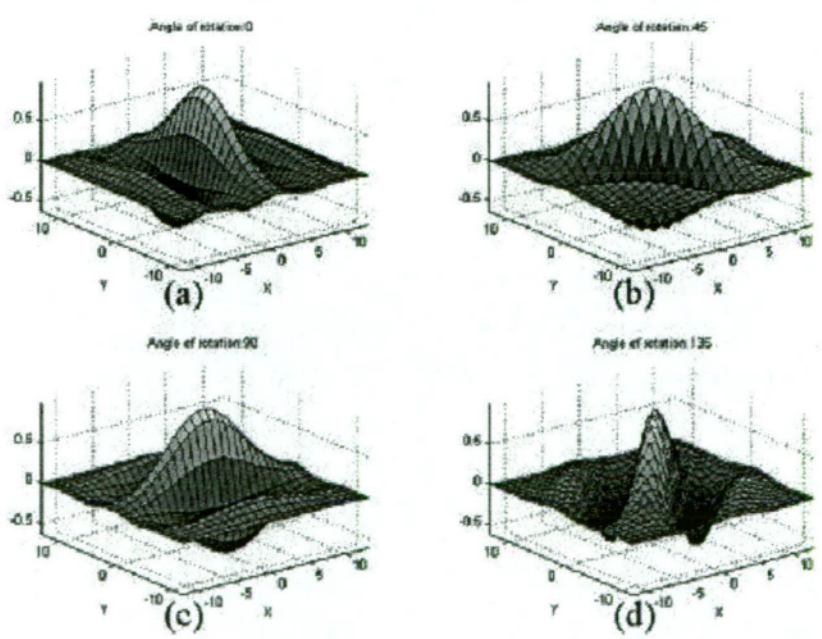


Figure 4.3: Response in 4 directions: (a) 0° , (b) 45° , (c) 90° and (d) 135°

4.8 Estimation of Gabor Filter's Parameters for Fingerprints

Fingerprints are characterised by ridge flow patterns that have two intrinsic properties: frequency and orientation. The Gabor filter is both frequency-selective and orientation-selective and can be tuned to corresponding local frequency and orientation of fingerprint for removing undesired noise. These properties are the reason why the Gabor filter is a suitable filter for filtering fingerprints. The parameters of the Gabor filter can be easily estimated by a visual inspection of the fingerprint but the corresponding automatic estimation is a non-trivial matter. The inter-ridge spacing (or Pitch) is estimated with respect to the resolution of the fingerprint, i.e.

$$P_\phi = \frac{N_p}{N_r}$$

where N_p and N_r is the number of pixels and the number of ridges, respectively, in the counting interval in the ϕ-direction. Figure 4.4 shows the ridge positions in the fingerprint over the estimating interval.

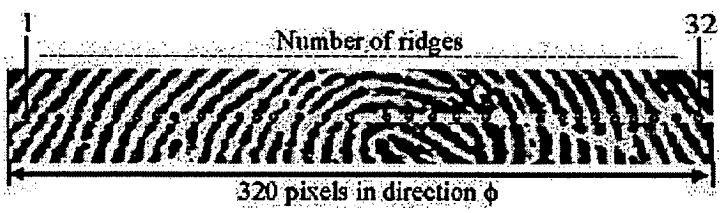


Figure 4.4: (•) Ridge positions.

The example in Figure 4.4 gives $P_\phi = 10$ pixels and hence u_ϕ is 1/10 cpp. We have chosen the standard deviation of the Gaussian function in x and y directions to be half of P_ϕ so that the Gaussian envelope covers at least a complete cycle of the signal (see Figure 4.1(c)). In Chapter Three we will describe an automatic ridge frequency estimation algorithm that is used in our fingerprint processing.

4.9 Filtering Fingerprints using Directional Gabor Filter in Spatial Domain

The 2-D directional Gabor filter in the spatial domain can be used for filtering fingerprint images. It enhances ridges that are oriented to the same angle as the filter while suppressing ridges that are oriented perpendicular to the filter's direction. In

Figure 4.5 the filtered fingerprint is calculated in spatial domain using the following parameters:

ϕ: 0°, 45°, 90° and 135°

u_o: 1/10 cpp

σ_x: 5 pixels

σ_y: 5 pixels

image size: 360 x 320 pixels

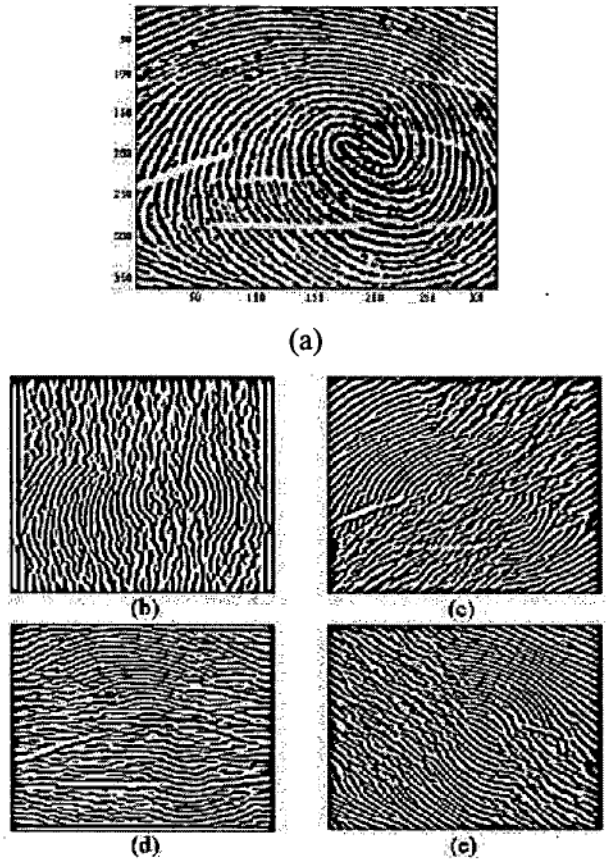


Figure 4.5: Results of filtering with 2-D directional Gabor filter: (a) original image, (b) filtered at 0°, (c) at 45°, (d) at 90°, and (e) at 135°.

Ridges are filtered along corresponding frequency and direction of the filter as shown in Figure 4.5. A simple visual inspection can be used to confirm the enhanced clarity of ridges flowing in the direction of the applied Gabor filter.

4.10 Two-Dimensional Directional Gabor Filter in Spatial-Frequency Domain

The Gabor filter in spatial-frequency domain in the closed form is obtained by taking the Fourier transform of (4.7), i.e.

$$H(u,v;\phi,f) = \int_0^{\infty} G2D(x,y,u_o,\sigma_x,\sigma_y,\phi)\exp\{-j2\pi(ux+vy)\}dxdy$$

By writing $\cos(2\pi u_0 x) = \frac{1}{2}[\exp(j2\pi u_0 x) + \exp(-j2\pi u_0 x)]$, the integral above can be reduced to

$$\begin{aligned}&\int_0^{\infty}\exp\{-[\frac{x^2}{2\delta_x{}^2} + j2\pi(u-u_0)x]\}dx\int_0^{\infty}\exp\{-[\frac{y^2}{2\delta_y{}^2} + j2\pi vy]\}dy + \\ &\int_0^{\infty}\exp\{-[\frac{x^2}{2\delta_x{}^2} + j2\pi(u+u_0)x]\}dx\int_0^{\infty}\exp\{-[\frac{y^2}{2\delta_y{}^2} + j2\pi vy]\}dy\end{aligned} \qquad (4.8)$$

From tables of integration, e.g. Gradshteyn and Ryzhik [11], under the form

$$\int_0^{\infty}\exp\{-p^2x^2 + qx\}dx,$$

it can be proved that (4.8) gives

$$H(u,v;\phi,f) = 2\pi\delta_x\delta_y\exp\left\{-\frac{1}{2}\left[\frac{(u-u_o\cos\phi)^2}{\delta u^2} + \frac{(v-u_o\sin\phi)^2}{\delta v^2}\right]\right\} \qquad (4.9)$$

$$+ 2\pi\delta_x\delta_y\exp\left\{-\frac{1}{2}\left[\frac{(u+u_o\cos\phi)^2}{\delta u^2} + \frac{(v+u_o\sin\phi)^2}{\delta v^2}\right]\right\}$$

Note that $H(u,v)$ is real because its spatial counterpart $G2D(x,y)$ in (4.6) is symmetrical. Figure 4.6 shows the plots of the 2-D Gabor filter in spatial frequency domain with σ_x = 5 pixels, σ_y=5 pixels, ϕ=0° and u_o=1/10 cpp.

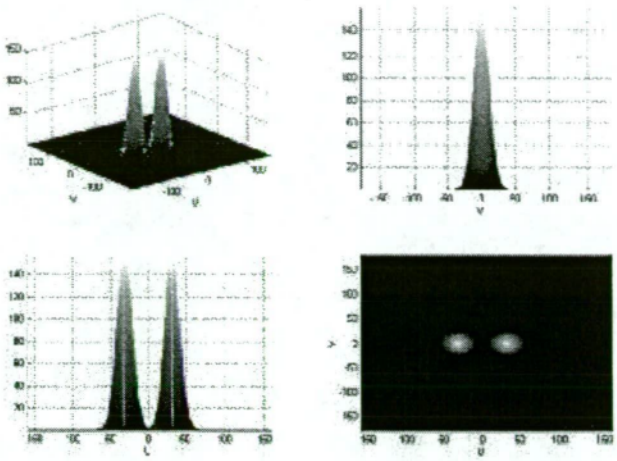


Figure 4.6: 2-D Gabor filter in the spatial-frequency domain in 0° direction.

Finally, the 2-D directional Gabor filter in the rotated spatial-frequency domain (u',v'), tuned to frequency u_0 and direction ϕ, is obtained by simply replacing (u,v) in (4.9) by (u',v'), calculated from the coordinate transformation

$$u' = u\cos\phi + v\sin\phi$$

$$v' = -u\sin\phi + v\cos\phi.$$

4.11 Filtering of Fingerprints using Gabor Filter in Spatial-Frequency Domain

It is well known that a convolution process is used in the spatial domain, while multiplication is used in the frequency domain for filtering. Therefore 2-D Gabor filtering in the spatial-frequency domain is less complicated than in the spatial domain. Equation (4.10) gives the equivalence between spatial and spatial frequency domain operations.

$$f(x) * g(x) \leftrightarrow F(\omega).G(\omega) \qquad (4.10)$$

Image filtering using the Gabor filter in the frequency domain has five steps as shown in Figure 4.7. Because the input and output format of the FFT routine is that the DC component is located at the start of the FFT array, i.e. indexed zero, spectral alignment is necessary for the multiplication in (4.10). The second step, therefore, shifts the image spectrum so that it is aligned with the double-sided spectrum of the Gabor filter in (4.9). The third step carries out filtering in the frequency domain by point-to-point multiplying. The fourth step shifts the spectrum of the filtered image back to the original form with DC at the bottom left of the spectrum ready for the

inverse 2-D fast Fourier transform (IFFT). As an example the above processing steps are applied to a fingerprint image using a Gabor filter in the frequency domain with the following parameters:

ϕ: 0°

u_o: 1/10 cpp

σ_x : 5 pixels

σ_y : 5 pixels

image size: 360 x 320 pixels

The result of filtering is shown in Figure 4.8.

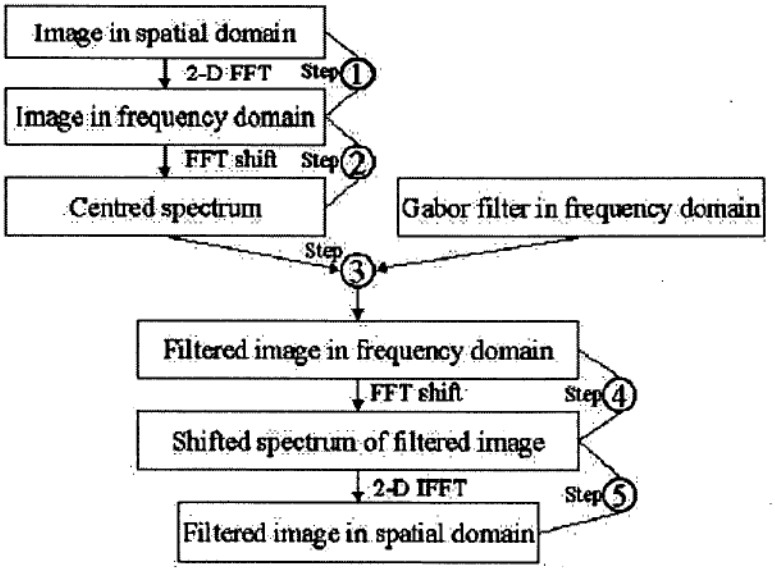


Figure 4.7: Processing steps in Gabor filtering in the spatial frequency domain.

Figure 4.9 shows a fingerprint image filtered using a 2-D directional Gabor filter tuned to four different angles ϕ and having the following parameters:

ϕ: 0°, 45°, 90° and 135°

u_o: 1/17 cpp

σ_x : 8 pixels

σ_y : 8 pixels

image size: 360 x 320 pixels

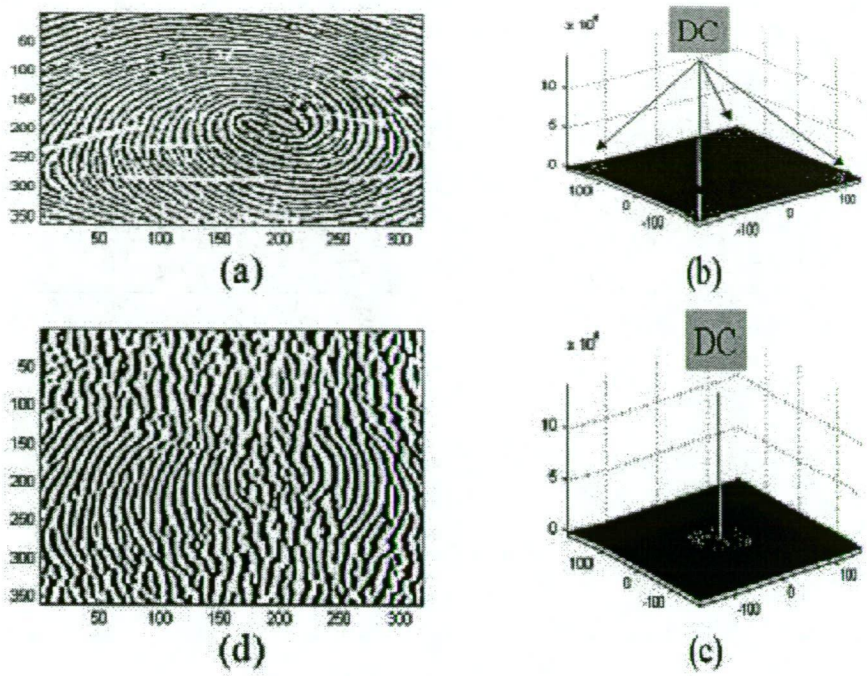


Figure 4.8: (a) Original image, (b) image spectrum before shifting, (c) image spectrum after spectrum shifting and (d) filtered image with enhanced vertical ridges.

4.12 Direction and Frequency Selective Gabor Filter Array

In this chapter, we propose to design a 2-D array of 8x4 Gabor filtered images $I_{ij}(x,y,\phi_i,u_{0j})$ for eight directions $\phi_i=22.5^0$ i, (i=1,2,...8), and four ridge frequencies $u_{0j} = 1/(3j)$, (j=2,...5). The range of j is estimated using the technique as described in Section 3.4. The spatial constants $\delta x= \delta y$ of filters tuned to u_{0j} are set to be equal to $1/(2\ u_{0j})$. The filtered image block at any combination of ridge orientation (as estimated from Section 3.2) and ridge frequency (as estimated from Section 3.3) can then be calculated by a 2-D interpolation. In this chapter, a simple 2-D separable linear interpolation technique that consists of a 1-D linear interpolation in ϕ, followed by a 1-D linear interpolation in u_0, is proposed. The 1-D linear interpolation in ϕ, and similarly for the interpolation in u_0, can be explained in Figure 4.10.

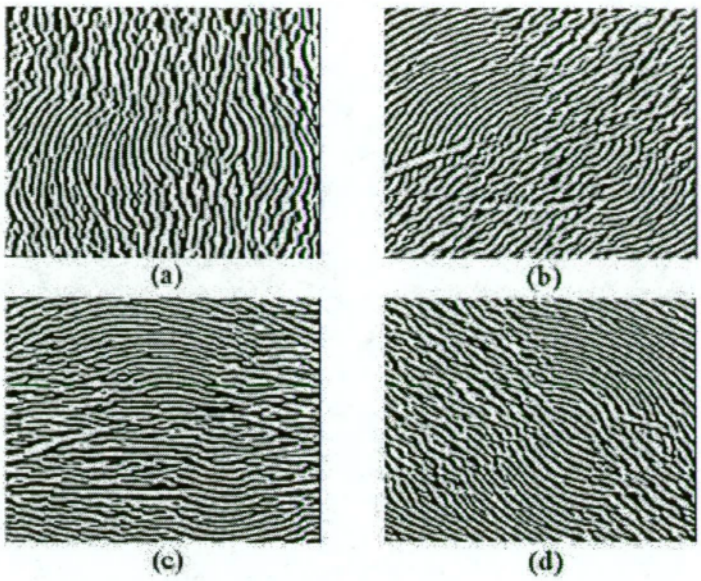


Figure 4.9: Results of a fingerprint filtered by a 2-D directional Gabor filter tuned to different directions: (a) 0°, (b) 45°, (c) 90° and (d) 135°.

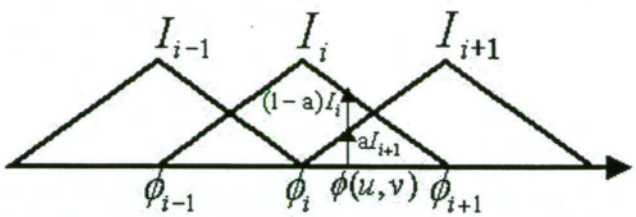


Figure 4.10: Linear interpolation of filter bank's output.

A response in any other direction $\phi(u,v)$ can thus be calculated by combining the outputs from the two adjacent filters as follows:

If

$$i = \text{Truncate}[[\phi(u,v)/22.5^0]\text{mod}8 + 0.5] \text{ and } a(u,v) = |\phi(u,v) - \phi_i|/22.5^0 \quad (4.8)$$

then

$$I_\phi(u,v) = a(u,v)\, I_{i+1}(u,v) + [1-a(u,v)]I_i(u,v), \quad \text{if } \phi(u,v) > \phi_i$$

but

$$I_\phi(u,v) = a(u,v)\, I_{i-1}(u,v) + [1-a(u,v)]I_i(u,v), \quad \text{if } \phi(u,v) < \phi_i \quad (4.9)$$

4.13 The Proposed 2-D Filtering Algorithm

Apart from a number of trivial image processing steps such as normalisation and histogram equalisation, the main steps in the proposed fingerprint image enhancement algorithm are illustrated in Figure 4.11 and are summarised as follows:

Step 1: Partition the fingerprint image into non-overlapping square blocks of size 8x8.

Step 2: Estimate the local ridge orientation for every block.

Step 3: Estimate the local ridge frequency for every block.

Step 4: Calculate the 8x4 array of directional filtered images, tuned to eight equally spaced ridge orientations and four equally spaced ridge frequencies.

Step 5: For each block centred at pixel (m,n) in the original image and with estimated ridge orientation θ and frequency u_0, identify the four blocks at the same location (m,n) in the four directional filtered images in the 8x4 array having direction and frequency on either sides of $\phi = (90^0 - \theta)$ and u_0 respectively, i.e. $\phi_i < \phi < \phi_{i+1}$ and $u_{0j} < u_0 < u_{0,j+1}$.

Step 6: Use the 2-D linear interpolation technique as described in Section 4.12, to calculate the enhanced image block from the four blocks identified in the previous step. Repeat Step 5 for the entire image.

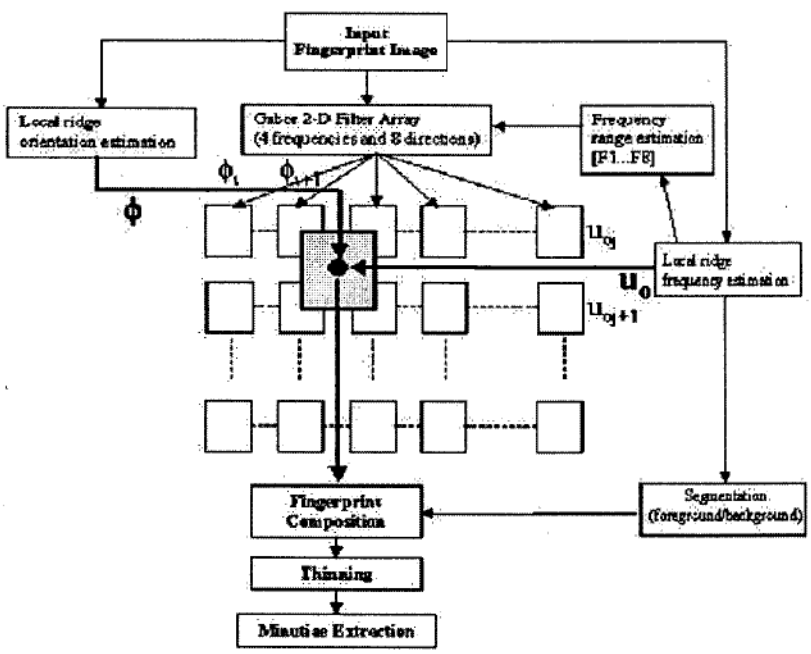


Figure 4.11: Steps involved in the proposed fingerprint image enhancement algorithm.

It is not easy for lay observers to visually spot differences between the two enhanced images in Figures 4.12(a) and 4.12(b). But poor resolution in 4.12(a) compared to 4.12(b) in the boxed areas, as an example, has caused a number of minutiae in 4.12(a) not to be detected (i.e. missed). In the top box, the two minutiae are simply too close

to each other to be detected. Overall, Figure 4.12(a) appears to be more 'regular' than 4.12(b) because the Gabor filters have been tuned to the same frequency.

(a) (b)

Figure 4.12: (a) Enhanced image with u_0 fixed to 1/8 [3] and (b) Enhanced image from our algorithm [4].

4.14 Conclusion

This chapter has provided the background on filtering of a fingerprint using a 2-D directional Gabor filter to perform directional filtering of fingerprints, both in spatial and spatial frequency domains. The filter can be tuned to appropriate frequency and orientation for enhancing the ridge pattern and removing undesired noise and distortion in fingerprints. The result of a directional Gabor filtering process is usually in the form of an array of images filtered in equally spaced directions. Therefore the filtered images cannot be used separately and must be combined into a single enhanced image before further processing is applied. Existing enhancement algorithms [3,4,8] usually have to combine Gabor filtering with other processing techniques to obtain satisfactory improvement of fingerprint images. Gabor filtering, however, is by far the most effective and most mathematically elegant fingerprint enhancement step. The new approach of our fingerprint enhancement algorithm employs both orientation and frequency of ridge in order to give the appropriate parameters estimated from fingerprint to Gabor filter. The result in Figure 4.12 has shown the structure of ridge that can be mostly retained using visual inspection. In chapter 6, the goodness index will be used for assessing performance of fingerprint image enhancement between current techniques and the proposed technique.

Chapter Five: Segmentation of Fingerprint Image

5.1 Introduction

Image segmentation is a common term for a variety of image operations. The image segmentation is a necessary step in any image-processing task. The simplest form of image segmentation splits the image into two parts, the foreground and background. In this chapter, segmentation is applied to an enhanced fingerprint image in order to isolate regions between ridge/valley and other regions. This technique employs the variance levels of x-signature [8] (see 3.3) that give an obvious difference of levels between background and foreground.

5.2 Segmentation

Regions of well defined ridges/valleys are defined as foreground, and other regions in which ridges are smudged or not visible are defined as background. Therefore, if the variance of x-signature is calculated for both foreground and background, the foreground will have a higher level of variance than the background. A simple threshold technique can be used to separate or to classify the two regions. Let T be the threshold level and V_m be the mean of variance of x-signature. We chose

$$T = \frac{V_m}{2}$$

A binary mask is then created in the following rule:

if f(u,v) > T

    binary mask is labeled as foreground (binary value=1).

otherwise

    binary mask is labeled as background (binary value=0).

An example of a binary mask applied for enhanced fingerprint in Figure 5.2 is shown in Figure 5.1.

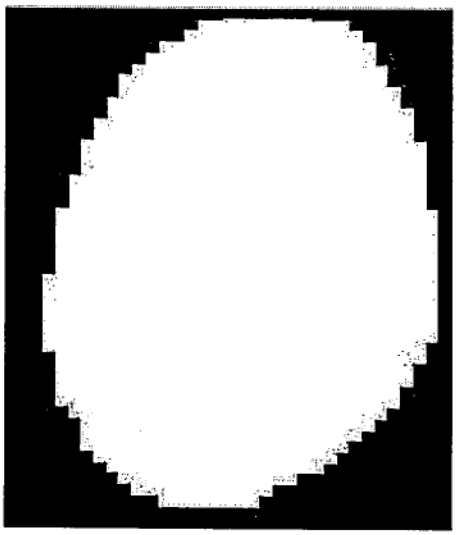


Figure 5.1: Binary mask.

The foreground and background are represented in white and black respectively. The binary mask is used to mask out the background of the enhanced image giving its foreground only as shown in Figure 5.2 (b).

(a) (b)

Figure 5.2: (a) Original image and (b) Masked enhanced image as foreground (in ridge region) and background (in white).

A casual visual inspection can ascertain the obvious segmentation in foreground/background between Figure 5.2(a) and Figure 5.2(b). The original fingerprint image has been corrupted with broken ridges and varying brightness along the ridges improved by the enhancement process.

5.3 Conclusion

The segmentation technique presented in this chapter has the ability to accurately segment regions between foreground and background in a fingerprint image. This technique uses a simple method for determining the threshold level of segmentation using the mean of the variance of the x-signature. For very noisy fingerprints, this is a very difficult task because the noise in the background region gives a similar variance level as in the foreground in some regions. The segmentation technique is then rarely able to separate the regions between foreground and background.

Chapter Six: Performance Assessment of Fingerprint Enhancement

6.1 Introduction

This chapter presents assessing performance of fingerprint enhancement using the goodness index with techniques such as thinning and minutiae extraction. After enhancing an image, the enhanced image is thinned using mathematical morphology and then the thinned image is employed for extracting minutiae points. The minutiae points extracted from the fingerprint are used for calculating goodness index values (GI). These values are represented levels of performance of the enhancement algorithm. For an ideal result, the GI value is equal to one (no error).

This chapter is organised as follows: Part 2 briefly explains the morphological processing of binary image as erosion, dilation, opening and closing, and then the thinning process applies hit-and-miss transform for thinning the binary image. The thinned image is used for extracting line endings and line intersections from which minutiae can be determined. Part 3 describes the minutiae extraction. Part 4 explains how detected minutiae points are used to assess the performance of a fingerprint enhancement algorithm by using the goodness index.

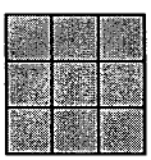
6.2 Morphological Processing of Binary Image

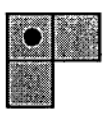
Morphological image processing is the most popular approach for thinning a binary image. There are two basic operations - *dilation* and *erosion* - and the three most common combinations of these two basic operations are *opening*, *closing* and *hit-and-miss* transformation. *Hit-and-miss* transformations use these basic operations to detect specific pixel configurations in a binary image. In fingerprint identification applications, such specific configurations are the ridge endings and ridge intersections or bifurcations. Different patterns of pixels within a small mask define different rotated structuring elements [13]. These elements are employed to scan over a binary image to produce conditions of 'hit-and-miss'. A 'hit' means the pattern of pixels of the mask and that of the binary image match each other - the corresponding output

pixel is set to one. A 'miss' means the patterns are mismatched and the output is set to zero.

6.2.1 Basic Morphological Operations

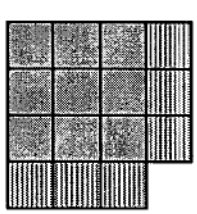
The four basic operations: erosion, dilation, opening and closing operating on set A by the structuring element B, are briefly described below:


A


B

Dilation: Dilation is an operation that results in an expansion of a binary object. The dilation of A by B, denoted as $A \oplus B$ in (6.1), is the set of all x displacements, such that if B and A overlap by at least one element, then entire elements of B are joined with A.

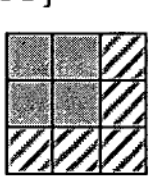
$$A \oplus B = \{x / x = a + b, a \in A, b \in B\} \qquad (6.1)$$

$A \oplus B =$ 


is an element added.

Erosion: Erosion is an operation that shrinks or thins a binary object. The erosion of A by B, denoted as $A \Theta B$ in (6.2), is the set of all x displacement, such that if –B and A non-overlap by at least one element, then the element of A at origin of -B is eliminated. –B is the 180° rotation of B around the origin.

$$A \Theta B^r = \{x / x - b \in A, b \in B\} \qquad (6.2)$$

$A \Theta B^r =$ 


is an element eliminated.

Opening: This function is used for smoothing the contour of an object by eliminating short and thin connections or spikes on the contour. The opening of A by B is denoted as $A \circ B$ in (6.3).

$$A \circ B = (A \ominus B) \oplus B \tag{6.3}$$

i.e. an erosion followed by a dilation.

Morphological opening completely removes regions of a binary object A that cannot contain or fit the given structuring element B.

Closing: This function operates in the same way as the opening function, i.e. for smoothing the contour of an object. However, unlike opening, it generally joins narrow breaks, fills long thin gulfs and fills holes that are smaller than the given structuring element. The closing of A by B is denoted as $A \bullet B$ in (6.4).

$$A \bullet B = (A \oplus B) \ominus B \tag{6.4}$$

i.e. a dilation followed by an erosion.

6.2.2 Hit-and-Miss Transformation

A powerful tool of mathematical morphology for shape detection in a binary image is the hit-and-miss transformation. The structuring element in this transformation is an element pair $B=(B_1,B_2)$, rather than a single element. The erosion of image A with structuring element B_1 determines the 'hit' structural features while the erosion of the complement of A with B_2 determines the 'miss', i.e. the complement features. Thus the intersection of the two erosion outputs gives the locations of the desired object shape. We shall describe this transform in the following:

Step 1: Define sets A, B and W as shown in Figure 6.1.

Step 2: Apply an erosion to set A using the structuring element $B_1=B$.

Step 3: Complement set A.

Step 4: Apply an erosion to the complement of A using the structuring element $B_2=W-B$ as shown in Figure 6.1.

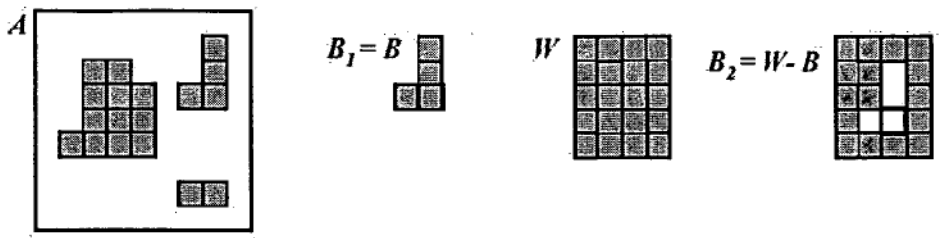


Figure 6.1: Sets A, B and W with defining sets B1 and B2.

Step 5: Take the intersection of the two erosion results from Step 2 and Step 4. The hit-and-miss transformation, denoted $A \otimes B$, is defined in (6.6). The result is shown in Figure 6.2.

$$A \otimes B = (A \Theta B) \cap \left[A^c \Theta (W - B)\right] \tag{6.5}$$

$$A \otimes B = (A \Theta B_1) \cap \left[A^c \Theta B_2\right] \quad if \quad B_1 = B \ and \ B_2 = W - B \tag{6.6}$$

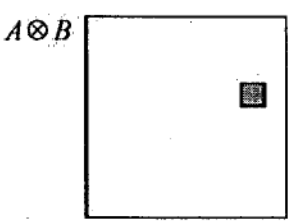


Figure 6.2: Result of hit-and-miss transformation

6.2.3 Thinning

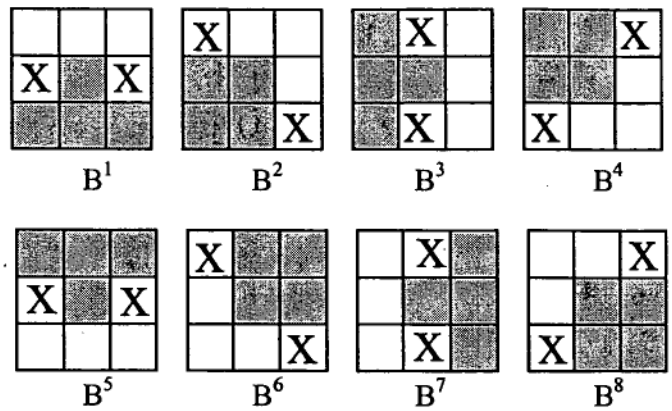
Thinning is a morphological image processing operation that applies hit-and-miss transformation to make thin lines from thick lines and shapes in a binary image. The thinning of a set A by a structuring element B in Figure 6.3 is denoted $A \Xi B$. In the following, we use an example taken from [6] to explain the thinning operation.

$$A \Xi B = A - (A \otimes B) \tag{6.7}$$

or $$A \Xi \{B\} = ((...((A \Xi B^1) \Xi B^2)...) \Xi B^n \tag{6.8}$$

where

$$\{B\} = \left\{B^1, B^2, B^3, ..., B^n\right\} \tag{6.9}$$



X: Don't care.

Figure 6.3: Set of rotated structuring elements.

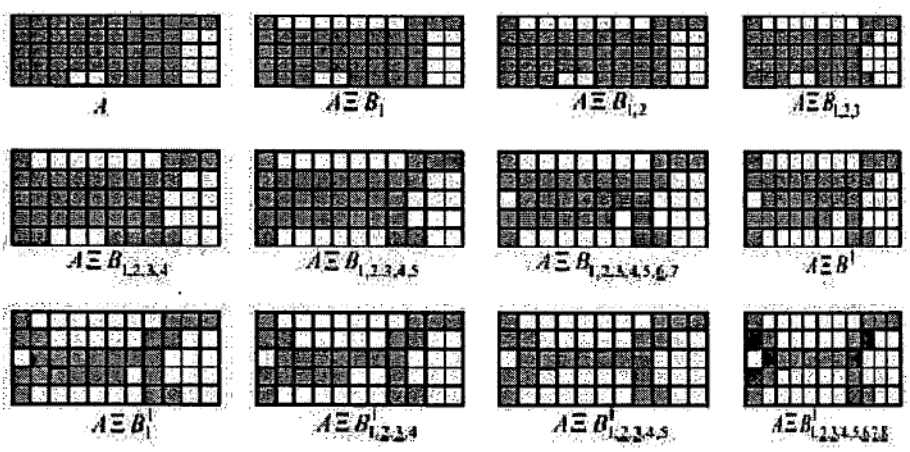


Figure 6.4: Set A thinned by successive application of structuring elements in Figure 6.3.

The process is to thin A by every pass of each structuring element in set B. This process is repeated until no further changes occur. The final thinned image is shown at the bottom right of Figure 6.4.

6.2.4 Detection of Line Intersection and Line Ending

Thinned images produced by the above thinning algorithm are not sufficiently robust for detecting the point of intersection of three lines and line endings, which is necessary for minutiae extraction. The thinned object usually possesses ambiguous doubling of elements or pixels. In [6] an 8-connectivity operation is used to further thin, i.e. eliminate doubling, the object from Figure 6.6(a) to Figure 6.6(c). In this section, however, we propose to use the rotated structuring blocks in Figure 6.5 as an

alternative for further thinning the same object. The improved thinned object is shown in Figure 6.6(b). If the object and the structuring block have more than two non-overlapping pixels, then the pixel at the origin of the block is eliminated. However, if two adjacent pixels in the object are eliminated then one of them is retained, as shown in Figure 6.6 (b). This process requires only one pass of each structuring block.

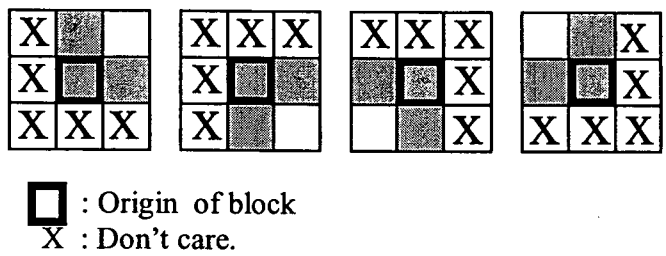


Figure 6.5: Rotated structuring elements used for further thinning.

The difference between the two techniques is not significant. From a simple visual inspection of the object in Figure 6.6(a), it is not clear which of the final objects in Figure 6.6(b) or in Figure 6.6(c) is more accurate. However, since the main objective of the processing in this section is to accurately detect the positions of ridge bifurcations and endings, the results of minutiae extraction, shown in Figures 6.8 and 6.9, prove that our technique serves as a satisfactory alternative to that in [6].

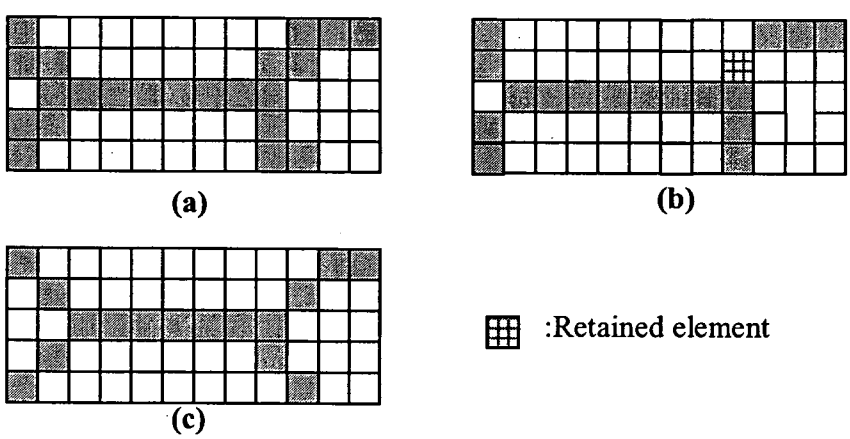
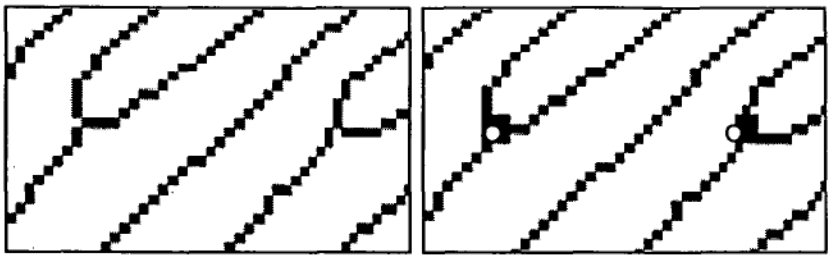


Figure 6.6: (a) Thinned image, (b) Thinned image improved by our technique and (c) Thinned image improved by m-connectivity [6].

An example of line crossing detection is shown in Figure 6.7.



(a) (b)

Figure 6.7: (a) Thinned lines, and (b) Marked locations of line crossings detected using the above procedure.

The location of a line ending can be detected using the same technique as described above for the detection of an intersection of three lines. When a line ending is encountered, the count of pixels in the image around the centre of the rotated structuring blocks is equal to one. The result of line ending detection is shown in Figure 6.8.

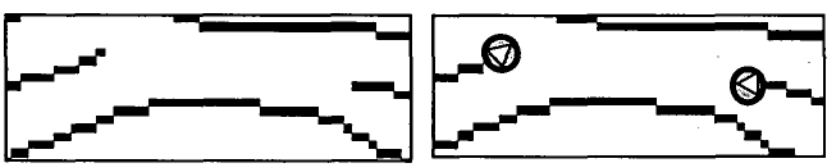


Figure 6.8: (a) Thinned lines with endings, and (b) Detected line endings, marked with white deltas.

6.3 Minutiae Extraction

Fingerprint matching techniques in automatic fingerprint identification systems are mostly minutiae-based. Minutiae are local features of the ridge pattern. Generally, regular ridges are parallel lines with gradually changing slopes, but minutiae are distinct features of ridges that do not follow the regular pattern. As many as 150 different local ridge characteristics (island, short ridges, enclosure, etc.) have been identified [8]. These types are not evenly distributed among people. The two most prominent ridge configurations, called *minutiae*, are *ridge bifurcation* and *ridge ending* [22]. Ridge bifurcation is a point at which a ridge splits or diverges into two ridges, and the ridge ending is a point at which a ridge terminates or ends. An example of minutiae is shown in Figure 6.9. Most fingerprint images of good quality contain about 40-100 minutiae.

Ridge bifurcation Ridge ending

Figure 6.9: Ridge bifurcation and ridge ending.

A thinned fingerprint image is required before a minutiae extraction algorithm [26,38] can be applied. Conventional algorithms usually use morphological block processing as described above in Section 6.2 for line intersections and endings. Angles and locations of ridge bifurcations and ridge endings are defined in Figure 6.10. These parameters are required for minutiae matching, and have to be calculated.

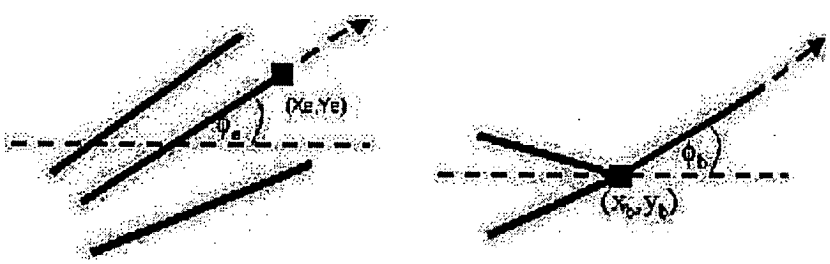


Figure 6.10: Line ending and crossing of lines

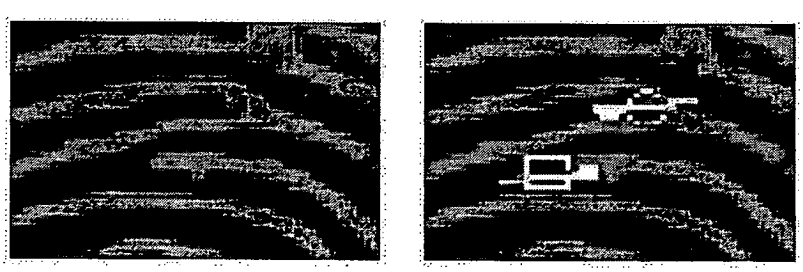


Figure 6.11: Minutiae: Ridge ending (circle) and ridge bifurcation (square).

In this chapter, the performance of the fingerprint enhancement algorithm used for minutiae extraction is assessed using a *goodness index* as defined in (6.10).

6.4 Results and Conclusions

The performance of our enhancement algorithm is judged against that in [8] in terms of (i) visual inspection of the ridge and valley structure, and (ii) reliability of minutiae detection. A more quantitative measure for the performance of a fingerprint image enhancement algorithm is the 'goodness index' (GI) of minutiae detection using the visually determined minutiae as reference. It is defined in [8,39] by summing over all b blocks as

$$GI = \frac{\sum_{i=1}^{b} q_i[p_i - m_i - s_i]}{\sum_{i=1}^{b} q_i v_i} \tag{6.10}$$

Since we treat all blocks as being equally important, the block quality factor $q_i = 1$, $\forall_i$.

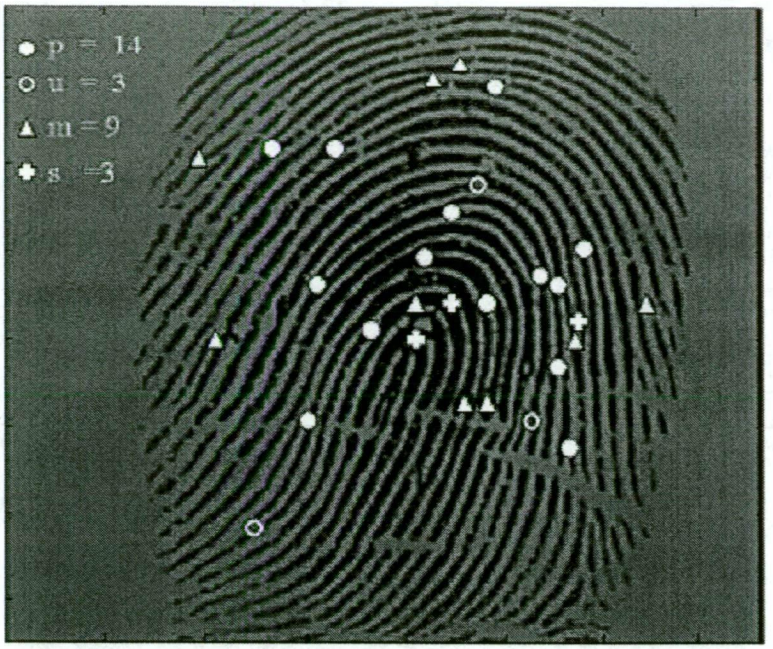
Therefore

$$GI = \frac{\sum_{i=1}^{r} (p_i - m_i - s_i)}{\sum_{i=1}^{r} v_i} \tag{6.11}$$

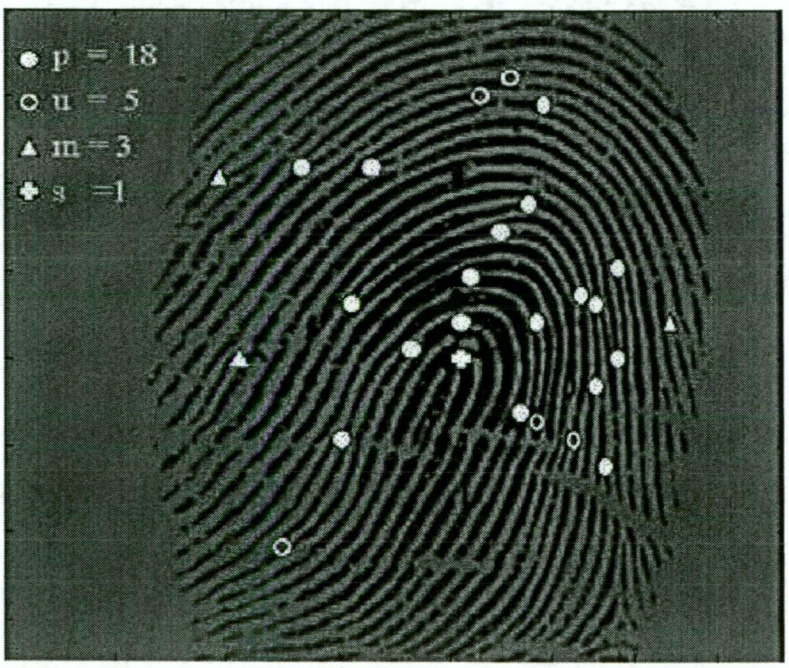
then

$$GI = \frac{p - m - s}{v} \tag{6.12}$$

in which p_i, m_i, s_i, and v_i are, respectively, the number of *paired* minutiae (i.e. detected with correct type), missed (i.e. undetected), spuriously (newly) introduced, and visually (manually) located minutiae in the ith block. Note that u_i, the unpaired (detected but with wrong type) minutiae are not used in the calculation of GI. In (6.10) we simply set the quality factor q_i = 1 for useable blocks and q_i =0 for discarded blocks. As an example, the total number of visually and manually determined minutiae in the fingerprint in Figure 6.12 is 26. It can be observed from Figures 6.13(a) that the algorithm using a fixed ridge frequency u_0 has many more missed minutiae. The GI for Figure 6.13(a) is 0.08 and for Figure 6.13(b) is 0.54. Of course, the ideal case is GI = 1.



(a)



(b)

Figure 6.12: Minutiae automatically detected from images (a) in Figure 4.12(a), and (b) in Figure 4.12(b).

Table 6.1 shows the GI values of seven fingerprint images randomly selected from the database in [10] and being enhanced using our enhancement algorithm.

Table 6.1: GI values of our enhancement algorithm when applied to seven different fingerprints.

Image No.	*v*	*a*	*p*	*u*	*m*	*s*	GI
1	20	20	18	1	1	1	0.8
2	39	36	28	7	4	1	0.6
3	25	26	23	1	1	2	0.5
4	46	42	37	2	7	4	0.6
5	35	39	26	4	5	9	0.3
6	41	40	28	5	8	7	0.3
7	24	24	19	1	4	4	0.5

The steps for assessing an enhancement algorithm can be summarised as shown in Figure 6.13.

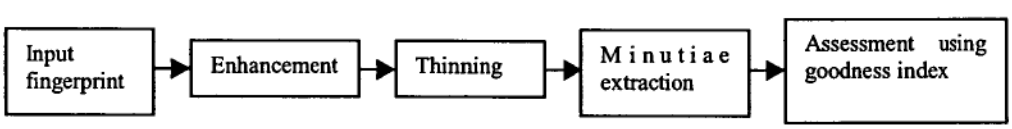
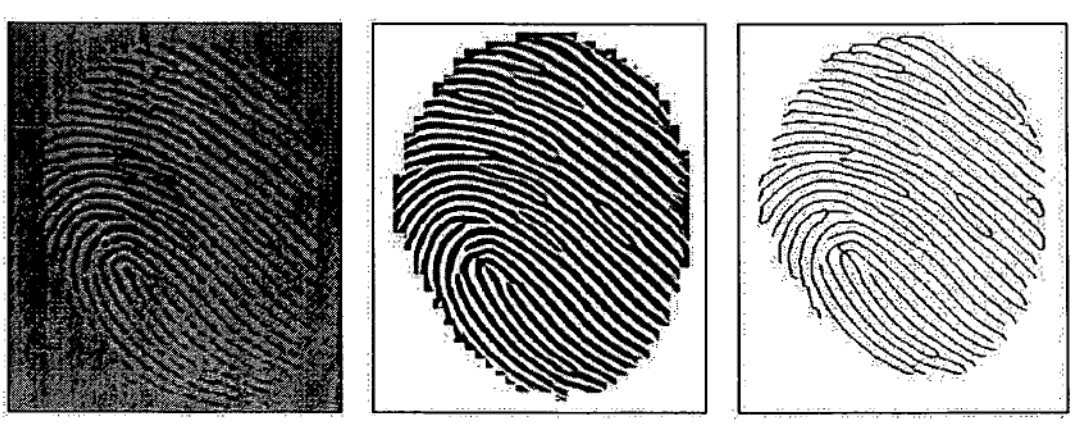


Figure 6.13: Steps in assessing the performance of a fingerprint enhancement algorithm.

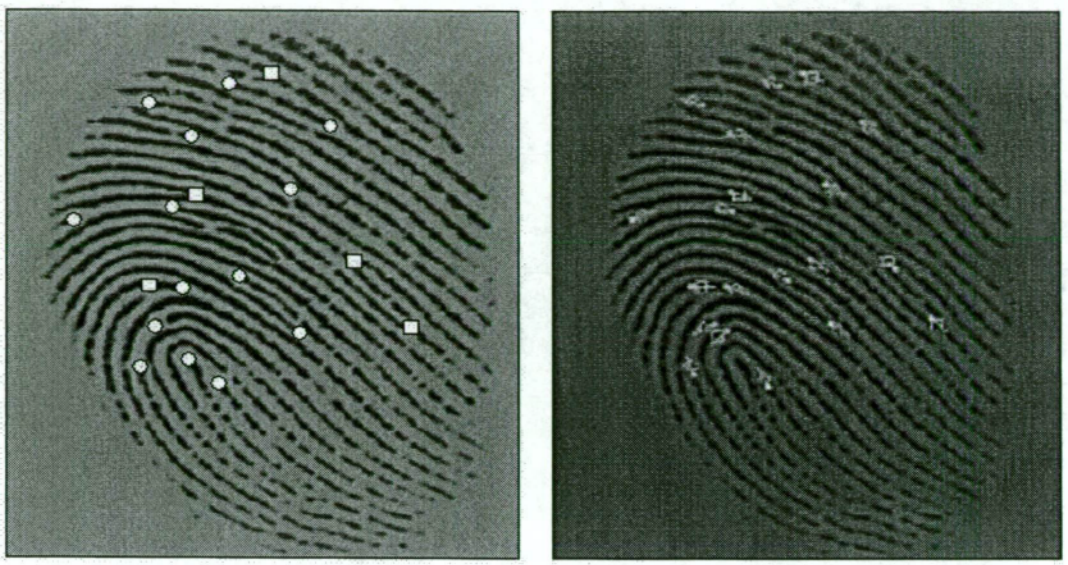


Original fingerprint Enhanced fingerprint Thinned fingerprint

Figure 6.14: Examples of input fingerprint and results of two processing stages before minutiae extraction.

Clarity of fingerprint is obtained by enhancement filtering. The enhanced image is then thinned so that it can be processed for minutiae extraction. After automatically

extracting the minutiae, the performance of the enhancement algorithm used can be assessed using a goodness index.



Visually counted minutiae Automatically detected minutiae

Figure 6.15: Extracted minutiae as ridge bifurcation ( ) and ridge ending (O).

Chapter Seven: Fingerprint Feature Extraction Techniques

7.1 Introduction

This chapter concerns only feature extraction techniques for characteristic features on a fingerprint, necessary for fingerprint classification. Minutiae, which are used for fingerprint identification, are not included in the classification stage. An accurate ridge-orientation estimation technique on a quincunx grid is proposed. The resulting double-resolution orientation map is then segmented into regions of *quantised ridge orientation*. A novel yet simple technique is proposed for the determination of the *singular points* (*SPs*) and the *region boundaries* [44] between different ridge flow orientations. The singular points and their associated *principal axes* are found to effectively characterise the ridge flow pattern and structure of a fingerprint for classification purposes, even when the finger impression is only partial. Our approach and that proposed in [14] both belong to a *structural* approach to fingerprint classification, in that both attempt to classify fingerprints using ridge flow patterns and structure, i.e. directional image segmentation, but is different in two aspects. Our segmentation technique is computationally much simpler, hence faster, and it also determines the position of singular points, necessary for the alignment required in the minutiae matching stage. In a recent paper [15], we also use the variance function of the local ridge orientation to determine the singular points and the flow region boundaries. The technique in [15] obviously incurs much higher computation costs.

The accuracy of the location of singular points on a fingerprint is important for minutiae matching alignment and is also essential for the Poincaré index to correctly determine the type of singular points. The technique proposed in this chapter offers a double-resolution estimation of the ridge orientation on a 4x4 pixel quincunx grid and quantises ridge orientations into six codes, called *ridge flow codes*. Singular regions are defined as those where *all six* codes exist. A singular point within a singular region can then be quickly determined by locating the point where the variance of local ridge orientation is at its maximum. The Poincaré indices of these singular points are used to determine their type: ordinary, delta, core or double-core. The

distribution and type of the singular points, together with their associated principal axes, may then be used to classify a fingerprint into one of six or seven well-known classes or patterns. That is the topic covered in Chapter 7.

The well-known *Henry*'s classification [40,41] scheme divides a fingerprint structure into three major classes or patterns: *Arch*, *Loop* and *Whorl*. Most people have loops but only a few have arches on their fingerprints. Researchers usually detail these three classes further into six or more subclasses or types based on finer differences within each class. Some composite types are combinations of the three *Henry*'s classes. In Figure 7.1 we distinguish seven classes of fingerprints commonly found in practice: Plain Arch, Tented Arch, Right Loop, Left Loop, Plain Whorl, Double Loop and Accidental Whorl (when the loop is rather irregular or not well defined). To test a classification technique, a large database of fingerprints is required and this is the subject of our further study.

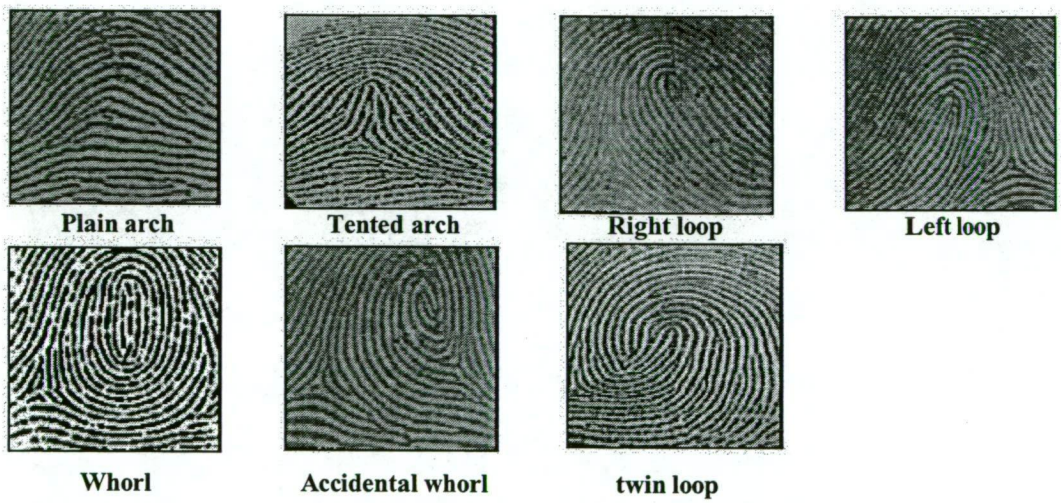


Figure 7.1: Seven classes of fingerprint commonly found in practice

This chapter is organised as follows: Part 2 is a review of feature extraction techniques for both global and local features. Part 3 proposes a new technique for improving the orientation field estimation on a quincunx grid on the fingerprint image. Part 4 proposes a new and simple approach for accurate determination of singular regions using ridge flow codes. Part 5 proposes two methods for locating singular points using ridge flow codes: one with the singular point being defined as the centre of a singular region, and the other with the singular point as the point of maximum orientation variance. Part 6 describes how to determine the type of a

singular point using the Poincaré index. Part 7 presents the results of rotation and scale invariant properties of singular points determined by our technique. Part 8 proposes a new technique for the determination of the principal axes of a core point, using an orientation mask.

7.2 Review of Feature Extraction Techniques

This review points to finger feature extraction for two types:

7.2.1 Global Features

Global feature is a feature that characterises the entire fingerprint image. This feature is more tolerant to noise and distortion, such as smudges and creases, than a local feature. This is due to the spatial randomness of noise and distortion that may occur in any region of a fingerprint. Local features such as minutiae in a small region are more likely to be affected by noise. Examples of two types of global features are described in the following sections.

7.2.1.1 Ridge Feature Map

Ridge feature map [18] of a fingerprint is the mapping of the intensity variance of the pixels within tessellated blocks. The fingerprint image is first filtered by eight equally spaced directional Gabor filters (at 0°, 22.5°, 45°, 67.5°, 90°, 112.5°, 135° and 157.5°) as in Figure 7.2. Each filtered image is divided into NxN non-overlapping blocks (Figure 7.3b) and the intensity variance of the ridges is then computed for each block as in Figure 7.3c.

Figure 7.2: Gabor filtered images in eight equally spaced direction [18].

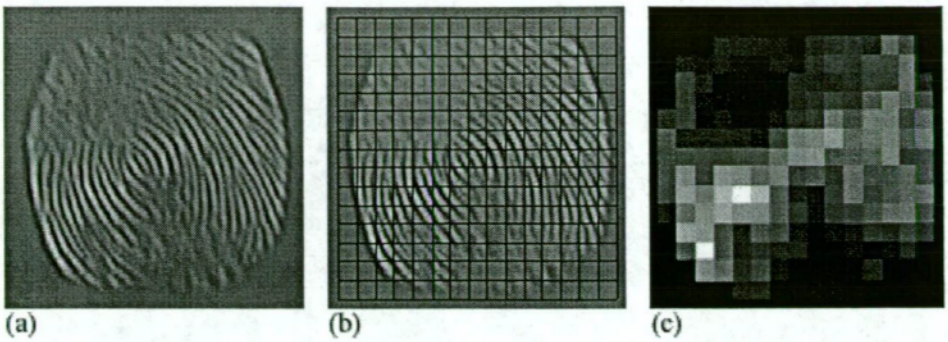


Figure 7.3: (a) Filtered image using Gabor filter at 157.5°, (b) Tessellated blocks on the filtered image, and (c) Ridge feature map of the filtered image [18].

7.2.1.2 Ridge Orientation Field

The ridge orientation field is a very commonly used block-based global feature of fingerprints. This is a mapping of the most dominant ridge orientation in a block. The details for the estimation of the most dominant ridge orientation in a block have been described in Chapter 3. An example is shown again here in Figure 7.4.

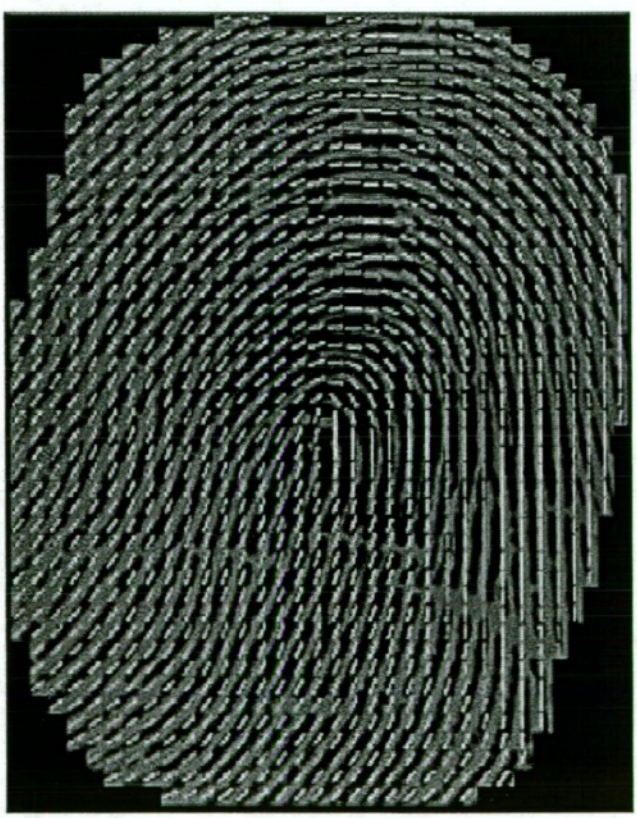


Figure 7.4: Estimated ridge orientation field superimposed on the fingerprint image.

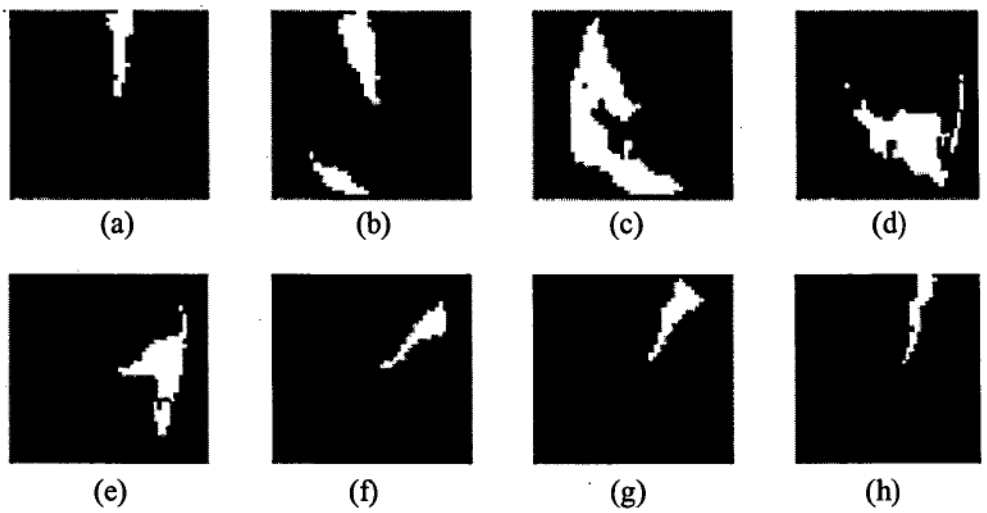


Figure 7.5: (a)-(h) Regions of the fingerprint in Figure 7.4 having eight orientation intervals equally spaced at 22.5°.

For visualisation, the distribution of ridge orientation on the fingerprint in Figure 7.4 is displayed in Figure 7.5. The regions are shown in white for eight equally spaced orientations.

7.2.2 Local Features

Local features such as singular points and minutiae are distinct features of individual ridges in a fingerprint. Local features are very sensitive to noise and distortion. It is therefore necessary to reduce noise and distortion using image enhancement techniques before reliable local feature extraction can be carried out. Since there are only one or two core-delta pairs in a fingerprint, singular points such as cores and deltas are not sufficient for fingerprint matching. However, they are the kernels of the global features that are used for classification. Minutiae are commonly used for fingerprint matching. Details of minutiae extraction method have been described in Chapter 6.

7.3 Quincunx Gradient Field Estimation

Since the ridge orientation (Section 3.2) has an 180^0 ambiguity, all estimation and filtering processes have to be done on the gradient field and the final gradients are converted to ridge orientations at the end. In this chapter, we propose to double the

gradient resolution to 4x4 pixels by estimating the ridge gradients on a *quincunx* grid as shown in Figure 7.6. This is achieved by estimating two overlapping gradient fields both using the usual 8x8 pixels estimation block size on a rectangular grid, but one is shifted *diagonally* from the other field by $4\sqrt{2}$ pixels. The two gradient fields are then merged together into a single quincunx field of 4x4 pixels resolution. The computation requirement is thus only doubled. To complete the final 4x4 pixels rectangular gradient field it is necessary to estimate the gradients at the pixels halfway between the pixels on the quincunx grid at which the gradients are already known. For this interpolative estimation, we simply calculate the average of the four known values, and horizontal and vertical components *separately*. Finally, the gradient angles $-90^0 \leq \phi(u,v) < 90^0$ (from Equation (3.1)) are calculated from their horizontal and vertical components. The ridge orientations $0^0 \leq \theta(u,v) < 180^0$ are obtained simply by adding 90^0 to $\phi(u,v)$.

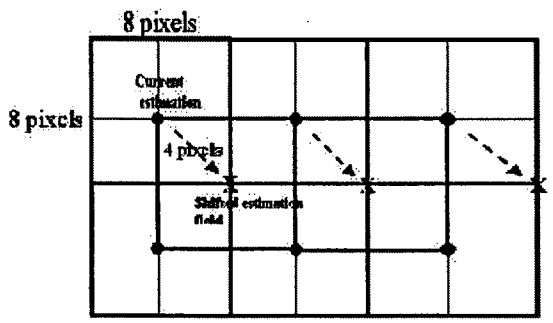
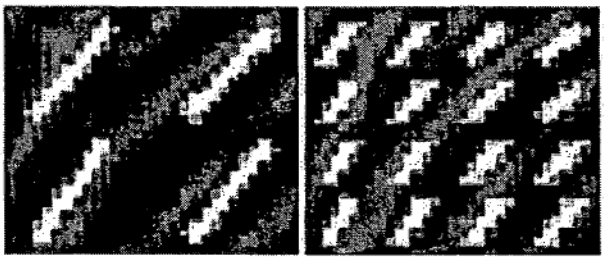


Figure 7.6: Gradient field on a 4x4 pixels quincunx grid

In Figure 7.7, we show a comparison between current techniques and our technique applying quincunx grid for estimating the ridge orientation. The white lines are the estimated dominant orientation while the true ridges are shown in black. The coarse estimation block size of 16x16 pixels used in many current techniques causes many parts of the ridge structure to be missed out as clearly shown for the bifurcation in Figure 7.7(a). In Figure 7.7(b), the bifurcation is correctly estimated and detected using our technique.



(a) (b)

Figure 7.7: Estimated orientation using (a) the current techniques, and (b) our technique.

7.4 Determination of Singular Regions Using Ridge Flow Codes

The range of ridge orientation estimated as described in the proceeding section is from 0 to 180^0 only because ridge orientation does not distinguish between two opposite directions. For fast determination of the location of singular points, we propose to quantise the ridge orientation in this range into six codes as explained in Figure 7.8. It is noted from the figure that because of the 180^0 ambiguity, orientations in the two ranges - between 0^0 and 14^0 and between 165^0 and 179^0 have the same Code 1. Thus each block is assigned a *ridge flow code* and a fingerprint is coded into a *ridge flow map* as illustrated in Figure 7.8 for an isolated right loop fingerprint.

As mentioned in the introduction, different classes of fingerprints have their own particular patterns of ridge structures. Global patterns are characterised by regions or segments of relatively uniform ridge orientation, i.e. having the same flow code. The boundaries between these regions can be detected and serve as global features of the fingerprint. Local patterns are characterised by localised *singular regions* in which ridge orientation changes sharply and in a complex manner. These regions are found by locating those blocks in the ridge flow map where *all six* codes exist or converge. *Singular points* are expected to be found within these singular regions. It is obvious that singular points can only be determined to within the resolution of a block size.

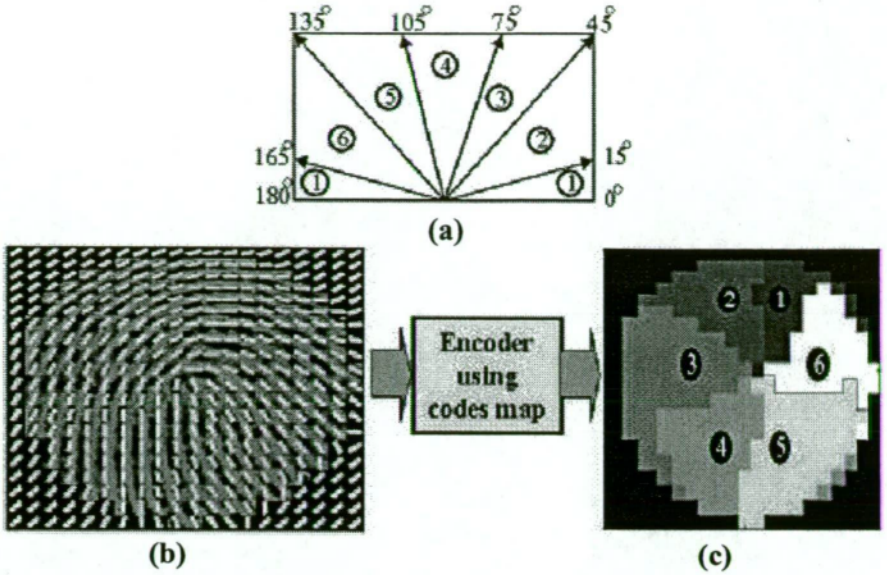


Figure 7.8: (a) Quantisation mapping for ridge orientation, (b) Ridge orientation field of an isolated right loop fingerprint, and (c) its ridge flow map.

Consider a region of interest of 7x7 blocks (i.e. 28x28 pixels). If the number of different codes found in the region is $p>1$, then the centre block is said to have *strength p*. The size of the region has been chosen to be approximately two inter-ridge distances so that it can cover all possible ridge orientations, i.e. six codes. The proposed algorithm for singular region determination is as follows:

Step 1: Scan the fingerprint with a 7x7 blocks window at *one*-block steps to calculate the strength of all blocks in the fingerprint. There may be more than one block of the same strength and they may be clustered closely together (i.e. connected).
Step 2: A singular region is defined as the cluster of connected blocks all having the maximum strength $p=6$. Determine the location of all the singular regions.
Step 3: If no p-code block is found, e.g. in an arch-class fingerprint, repeat Step 2 for $p = p-1$ until $p=2$. Otherwise, end the search.

The strength of a region of interest or point is a handy concept in dealing with a fingerprint class that has many variations in the global flow pattern but no singular points. There are many intermediate patterns between a 'gentle arch' and a 'peaky tented arch', depending on the strength of various flow regions. Note that the term 'singular' in this chapter refers strictly to region, block, or point of maximum strength $p=6$. The boundaries of the ridge flow regions, i.e. the global characteristics, can

easily be determined from the trajectories of the 2-code blocks. Figure 7.9 shows the result of the above algorithm for a tented-arch fingerprint. In Figure 7.9(c) the two singular regions (p=6) are clearly shown in white and the flow region boundaries are shown in grey lines.

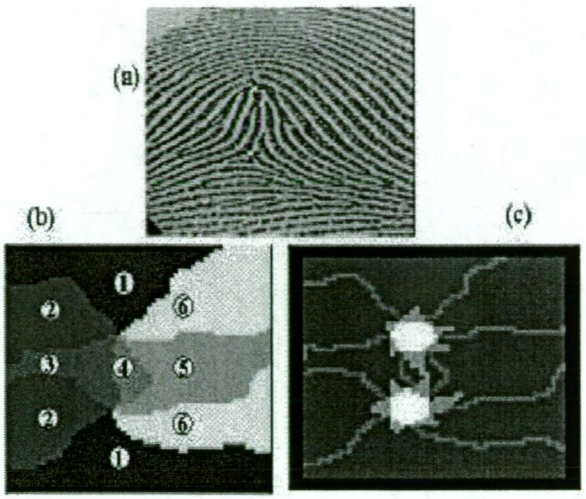


Figure 7.9: (a) Tented Arch fingerprint, (b) its ridge flow map, and (c) its singular regions p=6 (in white) and flow region boundaries p=2

7.5 Locating Singular Points on a Fingerprint

Once a singular region on a fingerprint has been determined, we *only* search this small region, usually about 10 blocks or fewer, for the corresponding singular point. We present below two techniques for locating the singular point of a singular region.

7.5.1 Singular Point as the Centroid of a Singular Region

In this technique, we define singular point as the centroid of a cluster of connected singular blocks. The (x,y) co-ordinates of the block centres, which are integers, are used to calculate the centroid's co-ordinates. The advantage of this technique is its simplicity. The main weakness is that the co-ordinates of the centroid have to be rounded to the nearest integers and the rounding error can be as much as 0.5 block. In the case of a closely situated core-delta pair, this rounding error can produce very drastic scale and rotation variances. The resulting directional error, as a simple rule, is $(0.5\text{x}180)/(D\text{x}\pi)$ degrees, where D is the distance between the core-delta pair

measured in number of blocks. The results of all six classes of fingerprints are shown in Figures 7.10-7.15.

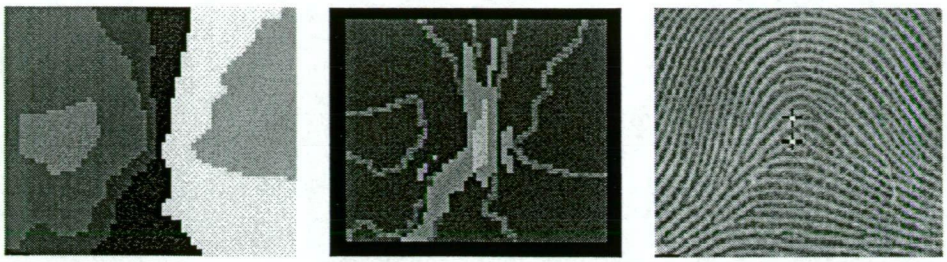


Figure 7.10: (a) Ridge flow map of a plain arch, (b) Singular regions p=3 (in white) and flow region boundaries p=2, and (c) Singular points of the plain arch marked by crosses.

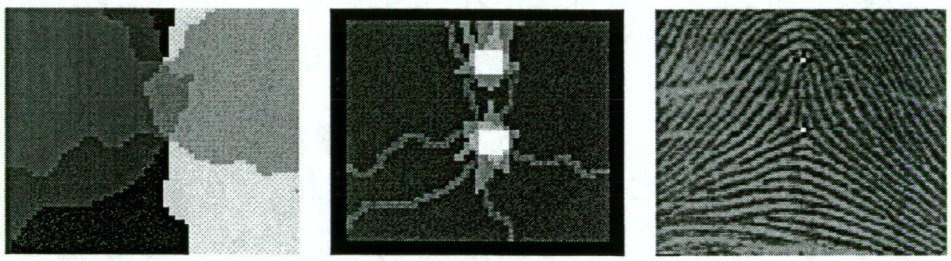


Figure 7.11: (a) Ridge flow map of a tented arch, (b) Singular regions p=6 (in white) and flow region boundaries p=2, and (c) Singular point of the tented arch marked by crosses.

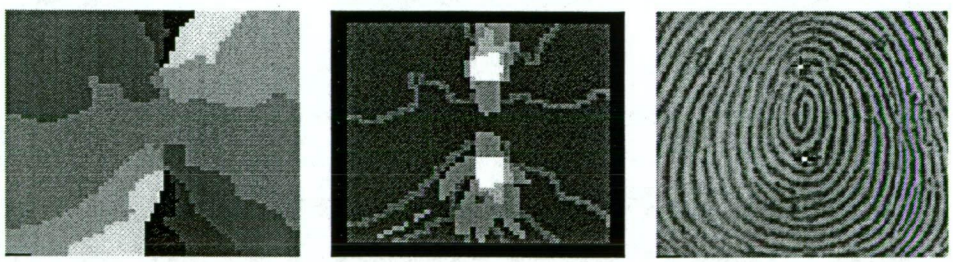


Figure 7.12: (a) Ridge flow map of a whorl, (b) Singular regions p=6 (in white) and flow region boundaries p=2, and (c) Singular point of the whorl marked by crosses.

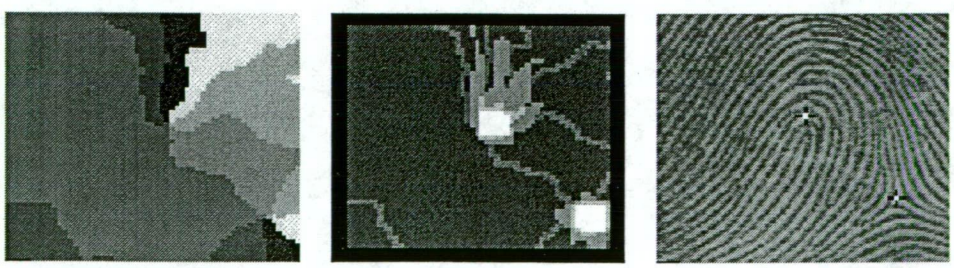


Figure 7.13: (a) Ridge flow map of a left loop, (b) Singular regions p=6 (in white) and flow region boundaries p=2, and (c) Singular point of the left loop marked by crosses.

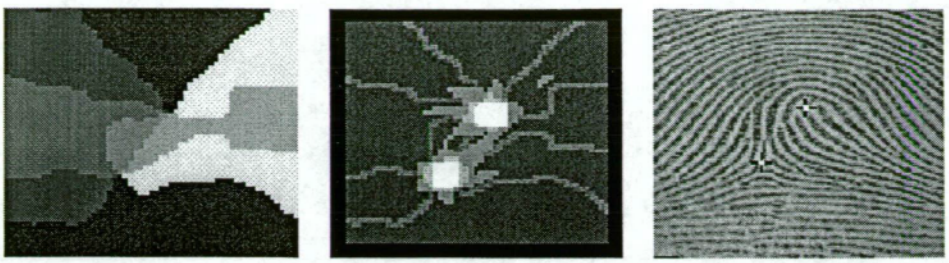


Figure 7.14: (a) Ridge flow map of a right loop, (b) Singular regions p=6 (in white) and flow region boundaries p=2, and (c) Singular point of the right loop marked by crosses.

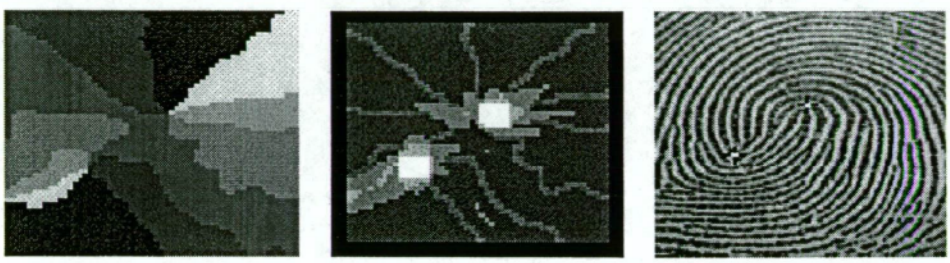


Figure 7.15: (a) Ridge flow map of a twin loop, (b) Singular regions p=6 (in white) and flow region boundaries p=2, and (c) Singular point of the twin loop marked by crosses.

This technique can quantify and detect the positions where the ridge flow pattern changes most on a plain arch as shown in Figure 7.10. A plain arch has no singular points, i.e. core or delta, and these locations of maximum variance should serve as useful reference points for the alignment of fingerprints in the matching process. The fingerprints as shown in Figure 7.11-15 have singular points in the regions where p=6.

7.5.2 Singular Point as Point of Maximum Orientation Variance

In [15], we make use of the *local* variance of the orientation to determine the locations of singular points. The local variance of the orientation at the centre of a block (x,y) is calculated over a region of 3x3 blocks centred on (x,y). Let $\nabla\theta_{p,q}(x,y)$ be the difference in orientation between the block (x,y) and its adjacent block (x+p,y+q), then to take care of the 180^0 ambiguity in θ we must choose the smaller of the two complementary angles below,

$$\nabla\theta_{pq}(x,y) = \min\left\{\begin{matrix}|\theta(x,y)-\theta(x+p,y+q)|, \\ 180^0-|\theta(x,y)-\theta(x+p,y+q)|\end{matrix}\right\}, \quad p,q = -1,0,+1 \tag{7.1}$$

The local variance of the orientation field at the block (x,y) is simply

$$\sigma^2_{\nabla\theta}(x,y) = \frac{1}{8}\sum_{p,q}\{\nabla\theta_{p,q}(x,y) - \overline{\nabla\theta_{p,q}(x,y)}\}^2 \qquad (7.2)$$

in which $\overline{\nabla\theta}$ denotes the mean value of $\nabla\theta$.

Table 7.1: Comparing co-ordinates of the singular points calculated by the two proposed techniques

		Centroid of singular region			Max. orientation variance		
		x	*y*	*Dist*	*x*	*y*	*Dist*
Tented Arch	-core	31	14	19.0	32	13	20
	-delta	32	33		32	33	
Right Loop	-core	33	19	35.4	33	19	35.8
	-delta	16	50		15	50	
Left Loop	-core	33	28	27.7	32	28	28.3
	-delta	51	49		51	49	
Twin Loop	-core	32	26	19.2	32	26	18.6
	-core	17	38		17	37	
Isolated	–core	31	17	19	31	17	18.0
Whorl	-core	31	36		30	35	
Whorl	-core	37	12	23.0	37	12	23.0
	-core	36	35		36	35	
	-delta	11	45		11	45	
Accidental	-core	42	10	19.0	42	10	18.0
Whorl	-core	48	28		48	27	
	-delta	53	36		53	35	
	-delta	12	44		11	44	

The technique in [15] requires the mapping of variance for the whole fingerprint and therefore is more computationally expensive than the ridge-flow code approach proposed in this chapter. An alternative to the centroid technique that avoids the rounding error in the location of singular points, is to define a singular point as the

centre of the block within the singular region, having highest orientation variance in (7.1).

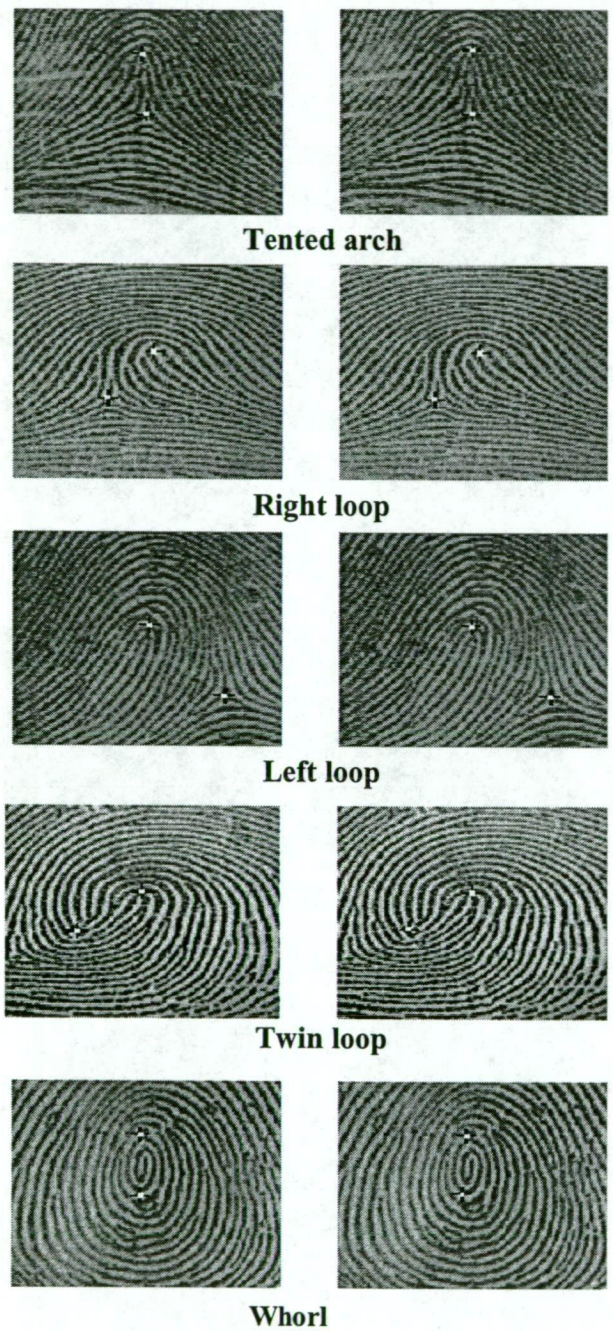


Figure 7.16: Marked singular points: (Left) located by centroid technique, and (Right) located by maximum variance of orientation technique.

Table 7.1 compares the co-ordinates (in number of blocks) of the singular points computed by the two different techniques proposed in Sections 7.5.1 and 7.5.2 above. For the classes of fingerprints in Figure 7.1, the co-ordinates of the singular points

given by the two techniques rarely differ by more than one block as shown on the left and right columns of Figure 7.16.

7.6 Determination of Type of Singular Points Using Poincaré Index

7.6.1 Types of Singular Points

Four common types of singular points have been identified by researchers, e.g. [5,28]: *Ordinary*, *Delta*, *Core* and *Double-Core*, named after their visual shape as illustrated in Figure 7.17.

Thus, a plain arch has neither cores nor deltas while a tented arch has a delta and a core. A loop also has a core plus a neighbouring delta, which usually is located at the bottom part of the finger and therefore is often missing from a partial finger impression as shown in Figure 7.18.

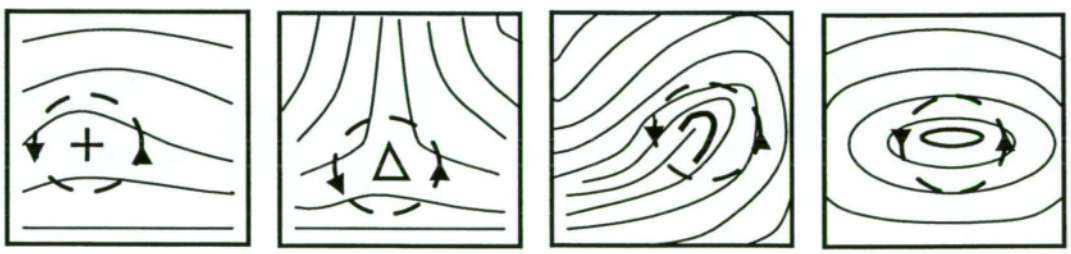
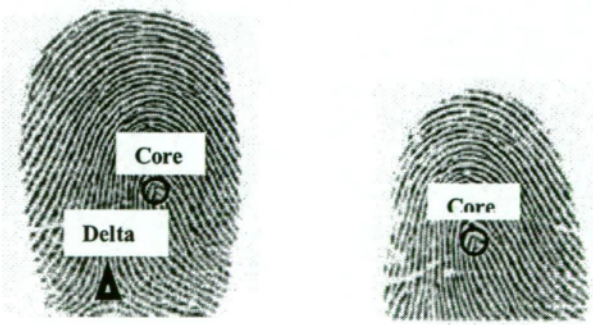


Figure 7.17: Definition of singular points and corresponding Poincaré indices



(a) (b)

Figure 7.18: (a) Right loop with core and delta, and (b) Partial fingerprint of right-loop type with missing delta.

7.6.2 Poincaré Index

Poincaré index is a measure of singularity in the trajectories of a dynamical system that can be described by differential equations [17]. Given a two-dimensional dynamical system in the (x,y) co-ordinates

$$\begin{aligned} \frac{dx}{dt} &= \phi(x,y) \\ \frac{dy}{dt} &= \psi(x,y) \end{aligned} \tag{7.3}$$

i.e

$$\theta = \tan^{-1}\left(\frac{\psi(x,y)}{\phi(x,y)}\right), \tag{7.4}$$

When the point (x,y) moves once around a closed contour C in the counter-clockwise direction (conventionally positive), the angle swept out by θ will be

$$\oint_C d\theta$$

The Poincaré index of the contour C with respect to the system in (7.3) is defined as

$$I(C) = \frac{1}{2\pi}\oint_C d\theta, \tag{7.5}$$

or in discrete form

$$I(C) = \frac{1}{2\pi}\sum_{i=1}^{N} \nabla\theta_i \tag{7.6}$$

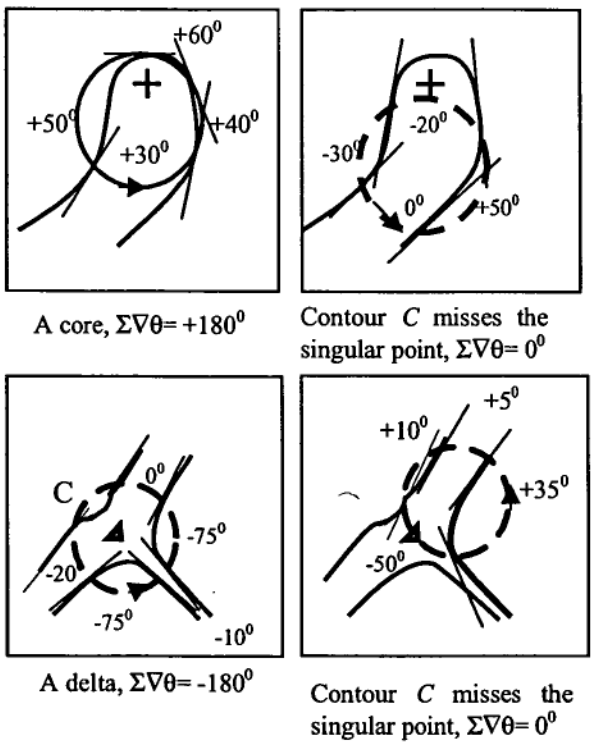
When applying the Poincaré index to the ridges of a fingerprint, because the line function has 180^0 ambiguity, we must choose the *smaller* of the two complementary angles for the difference between the two orientations. Therefore, we have similar to (7.1).

$$\nabla\theta_i = \min\left\{\begin{matrix} \left|\theta_{i+1} - \theta_i\right|, \\ 180^0 - \left|\theta_{i+1} - \theta_i\right| \end{matrix}\right\} \tag{7.7}$$

If $I(C) = 0$, the contour C does not contain any singular point in its interior.

If $I(C) = +1/2$, the contour C contains exactly one *core* singular point in its interior.

If $I(C) = -1/2$, the contour C contains exactly one *delta* singular point in its interior.



A core, $\Sigma\nabla\theta = +180^0$

Contour C misses the singular point, $\Sigma\nabla\theta = 0^0$

A delta, $\Sigma\nabla\theta = -180^0$

Contour C misses the singular point, $\Sigma\nabla\theta = 0^0$

Figure 7.19: Sensitivity of Poincaré index with the location of the summing contour.

Figure 7.19 exemplifies the importance of knowing accurately the location of the singular point in order to choose the summing contour C correctly. If C misses the singular point, its Poincaré index will be zero and the singular point will be interpreted as an *ordinary* point. Ideally, C is chosen to be the minimum enclosing contour of the singular point, otherwise it cannot separate two closely located singular points. For example, the Poincaré index of a contour enclosing a core and a delta together will be zero.

In [5,16], singular points are found by computing the Poincaré index of every orientation estimation block. The accuracy of the locations of the singular points found by their methods is therefore limited to within the estimation block size of 16x16 pixels in [16] and 8x8 pixels in [5] compared to 4x4 pixels in this chapter. Our approach in this chapter achieves a higher resolution for ridge orientation and a much simpler algorithm, hence much faster, singular point detection. This is because we have to compute the Poincaré index only for singular points to determine whether they are deltas, cores or ordinaries.

Once the location of a singular point has been found from the intersection of the flow codes as presented by the above algorithm, its type is then determined from the Poincaré index. If the computed index is 0, 1/2, -1/2 or 1, the singular point is an ordinary, core, delta or double-core respectively [5]. For classification purposes, we are interested only in the identification of the type of principal singular points. To obtain a valid Poincaré index for a singular point, the summing contour (see Figure 7.19) has to cut through at least two ridges. The size of the contour therefore, normally has to be proportional to the inter-ridge spacing. With the high accuracy produced by the proposed algorithm for the location of singular points, the Poincaré index almost always gives the correct type of singular points. In this chapter, we choose the summing contour C in (7.6) as the sixteen connected blocks in the outer of the 5x5 block neighbourhood centred at the singular block. It is obvious that this choice of C allows the co-ordinates of the enclosed singular point to have an error tolerance of ± 1 block.

Since ridges come in from one side and go out at the other side of a finger, we expect the overall Poincaré index of the whole fingerprint to be zero. Therefore, if the finger's impression or scanned image is complete, cores and deltas exist in pairs. Unfortunately, in practice finger print images acquired by optical or electronic scanners are not wide enough to cover all the deltas in the peripheries.

7.7 Rotation and Scale Invariance of Singular Point Determination Technique

We have pointed out in Section 7.5.2 that the chosen size of 5x5 blocks for the summing contour C allows an error tolerance of ± 1 block in the (x,y) co-ordinates of the singular point enclosed in C. Therefore, if rotation is moderate, say less than 20^0, the proposed technique is found to be scale and rotation invariant for all fingerprints in Figure 7.1. However under a large rotation, the ridge flow pattern is unavoidably distorted by the co-ordinate transformation process, and may result in significant error in the location of singular points. This is particularly the case for a very closely located core-delta pair as presented in Table 7.2.

Table 7.2: Result for a left loop under rotation

	Centroid of singular region			Max. orientation variance		
	x	***y***	***Dist***	***x***	***y***	***Dist***
Original -core	33	29	8.1	32	29	8.1
-delta	40	33		39	33	
15^0 rotation -core	31	29	8.2	31	29	8.2
-delta	39	31		39	31	
45^0 rotation -core	29	30	6.3	29	30	9.5
-delta	35	28		38	27	
	Ordinary			*Delta*		

It is observed from the table that for moderate rotations, the result is reasonably consistent in both techniques. However, at large rotations (which is rarely the case in practice) the discrepancy in the location of the delta becomes as much as three blocks in x and one block in y (last row of Table 7.2). This results in the delta being missed by the Poincaré index summing contour C and consequently, being wrongly detected as an ordinary in the centroid technique. The unavoidable distortion introduced by the rotation, also as expected, tends to create new blocks of equal strength p or even higher as shown in Figure 7.20.

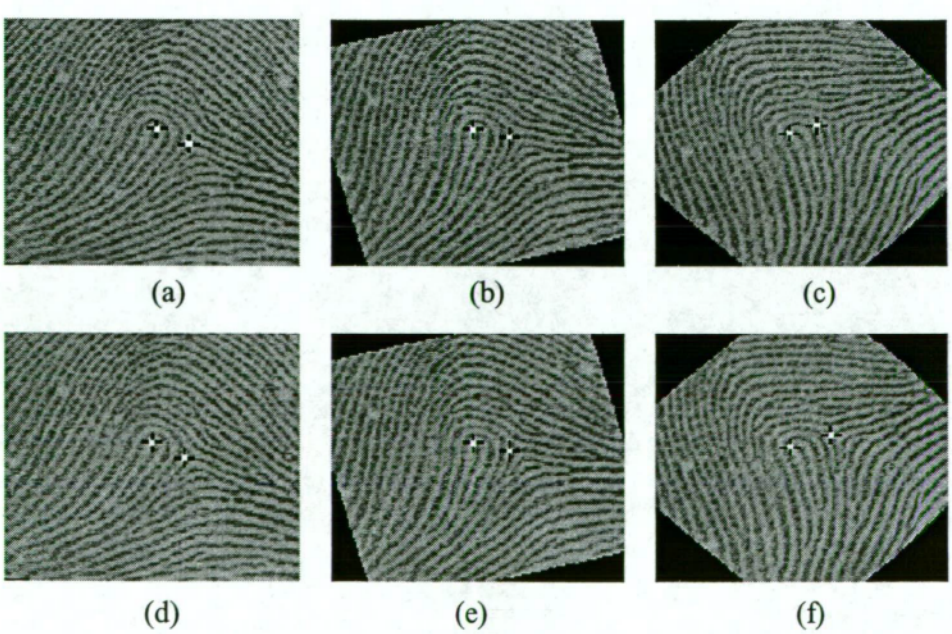


Figure 7.20: Singular points detected by (a)-(c) centroid of singular region technique under 0°, 15° and 45° rotation, respectively, and (d)-(f) maximum variance of orientation technique under 0°, 15° and 45° rotation, respectively.

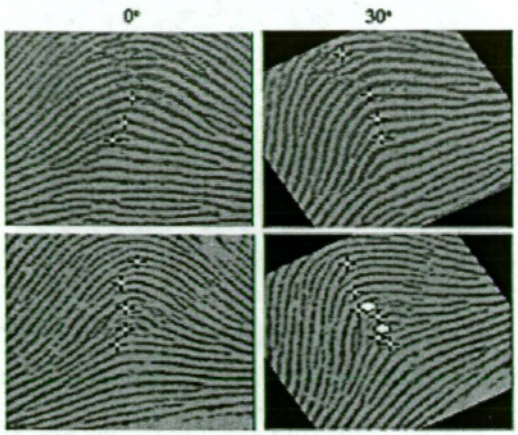


Figure 7.21: Blocks of strength p =3 (crosses) and p=4 (white dots) in an Arch class fingerprint, i.e. having no singular points.

This is evident in the case of arch class fingerprints in Figure 7.21 where the 30^0 rotation has produced more blocks of p=3 and two blocks of higher strength (p=4) shown by white dots. For this reason, it is important to track the *same* blocks when assessing the rotation invariance of the principal axis. This can be achieved if we *only* use a few blocks of highest orientation variance in the linear regression estimation of the axis, i.e. ignoring the 'weaker' blocks newly created by the rotation.

7.8 Determination of Principal Axes Associated with a Singular Point

Principal axes associated with a singular point are defined as the axes radiating from the point and having the dominant directions of the ridge flow near the singular point. The directional relationship between the principal axes and the axes joining the singular points, i.e. core-to-core and core-to-delta axes, effectively characterises the global structure of the fingerprint. Principal axes are therefore useful for distinguishing classes having the same number and type of singular points.

The topology of a loop, just outside the singular point, is characterised by the dominance of the almost parallel ridges in its 'tail', and by the almost semicircular ridges around the core at its 'head'. For a tented arch, the tail actually 'flares' out from the core and we can expect to have two dominant orientations. A loop that has a 'fat' head may have one more dominant ridge orientation apart from that of its tail, that has to be suppressed. We achieve this by using an *orientation mask* which

excludes the inner region around a singular point and suppresses *non-radial* orientations. The radial orientations are calculated from the block centres to the singular point. With the above observations, we propose the following algorithm:

Step 1: Consider an 'annular' region of size 15x15 blocks centred on a core and the excluded centre is 7x7 blocks.

Step 2: Subtract the orientation field in the annular region from that of a 15x15 block orientation mask.

Step 3: Threshold the difference orientations field by 45^0 and retain only those blocks with difference orientation smaller than the threshold. Figure 7.22(a) shows, only as an example, a small 9x9 block orientation mask with a 3x3 block skip-zone, designed for extracting dominant orientation of ridges around the core point of a right loop.

Step 4: Plot the histogram of the orientations θ of the blocks surviving from the thresholding in Step 3, where $0^0<\theta\leq 180^0$ using a bin-size of 10^0.

Step 5: Determine the dominant orientations (θ_{pi}, f_{pi}), i.e. the histogram peaks.

Step 6: If two adjacent peaks are separated from each other by only one bin-size, group them into one group keeping the middle component.

Step 7: If the adjacent component to a peak is stronger than *half* the strength of the peak, retain the component. This step together with Step 6 results in peaks θ_{pi} of width w_i, where $1\leq w_i \leq 5$ bin-sizes.

Step 8: Calculate the probability that orientation belongs to peak at θ_{pi}, which is

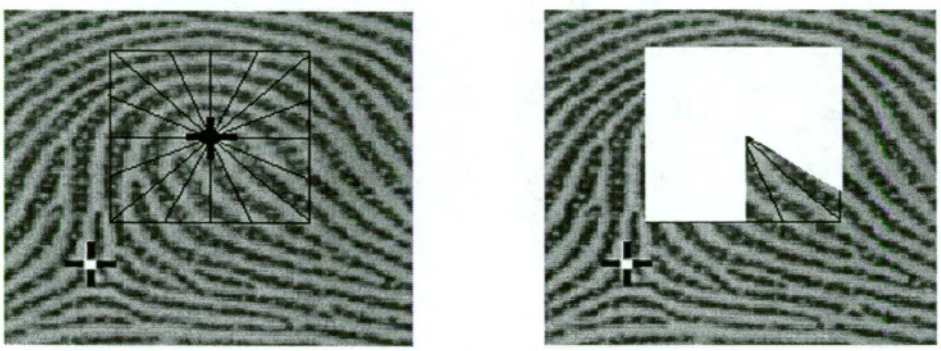
$$\Pr(\theta_{pi}) = \frac{1}{N}\sum_{k\in w_i} f_k \qquad (7.8)$$

It is important that w_i is kept the same for all peaks, i.e. the same number of components is summed up in (7.1).

Step 9: Retain only the peak with the highest probability and those with a probability equal or better than 30% of the highest probability. The surviving peaks give the estimate of the dominant orientations in the loop's tail.

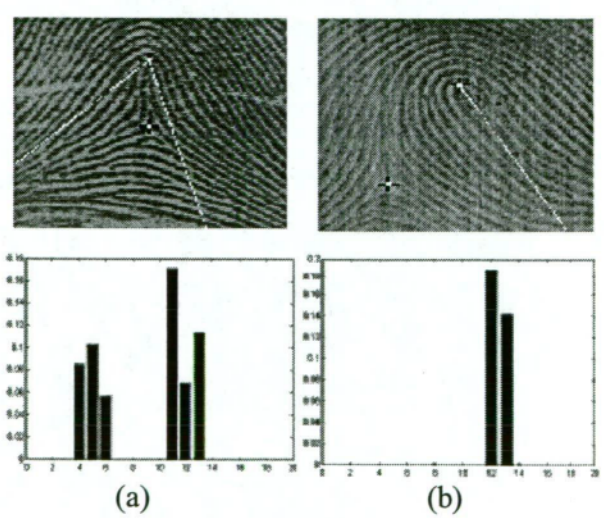
135	127	117	104	90	76	63	53	45
143	135	124	108	90	72	56	45	37
153	146	135	117	90	63	45	34	27
166	162	153	-1	-1	-1	27	18	14
180	180	180	-1	-1	-1	0	0	0
14	18	27	-1	-1	-1	153	162	166
27	34	45	63	90	117	135	146	153
37	45	56	72	90	108	124	135	143
45	53	63	76	90	104	117	127	135

(a)



(b) (c)

Figure 7.22: (a) A 9x9 block orientation mask with 3x3 block skip, (b) Orientation mask superimposed onto the right loop fingerprint at the core position, and (c) Masked region (in white) and surviving region (no fill) within the orientation mask.



(a) (b)

Figure 7.23: Estimated principal axes of (a) a tented arch, and (b) a right loop

Figure 7.23 gives the estimated principal axes of a tented arch and a right loop. The chosen loop is rather 'fat' having pronounced orientation around its head and hence is

prone to having a second dominant orientation axis. The proposed mask in Figure 7.22 has successfully eliminated that orientation from its histogram.

7.9 Conclusion and Discussion

We have successfully presented a simple but surprisingly accurate algorithm for the detection of singular points and the principal axes of the cores on a fingerprint. The coarsely quantised orientation field maps the fingerprint into a ridge flow code map from which singular regions can be identified as clusters of blocks where *all* codes exist. The singular point of each singular region can then be found rather quickly by restricting the search to within the singular region, for the block having maximum variance of ridge orientation. With the proposed double-resolution estimation technique for the ridge orientation field, the location of singular points can be accurately determined and the Poincaré index is found almost always to identify correctly the type of singular points. It was found that under moderate rotation, the distribution of singular points and their associated principal axes is both rotation and scale invariant to within an analysis block (e.g. 4x4 pixels). This is sufficient for classification and alignment of fingerprints, necessary for any accurate minutiae-based matching. We are currently investigating the use of singular points and their associated axes for fingerprint classification purposes.

Chapter Eight: Fingerprint Classification

8.1 Introduction

This chapter proposes a rule-based algorithm for classification of a fingerprint into one of six well-known classes: ordinary arch, tented arch, right loop, left loop, whorl and twin loop. The rules are formed using the relative location and types of singular points and the relative directions of their associated principal axes. The reliable and fast classification algorithm is made possible by a simple but effective combination of ridge *flow-code* technique and *orientation variance* calculation in determination of singular points and principal axes. The Poincarè indices [42,43] of these singular points are used to determine their type: ordinary, delta, and core: the three types of singular points mostly used in current techniques [5,14,19,28] and in our algorithm for the classification process. While distinct patterns of a whorl and a twin loop are combined into one class in current techniques [5,14,19,28], our algorithm can distinguish between a whorl and a twin loop by using the relative position of the principal axis with respect to the two singular points. Since deltas and cores exist in delta-core pairs, we can also expect to be able to classify incomplete fingerprints in which deltas are not included. This occurs in many practical cases because cores are mostly found in the centre region while deltas are usually located in the outer region of a fingerprint.

This chapter is organised as follows: Part 2 briefly studies current fingerprint classification techniques. Part 3 explains our fingerprint classification algorithm using singular points and principal axes. Part 4 presents the experimental results and conclusions.

8.2 Current Fingerprint Classification Techniques

8.2.1 Singular Point Approach 1

An approach to fingerprint classification based on singular points has been proposed in [5,28]. These published papers concern mainly the coarse-level classification. The technique proposed in [5] can be described as having three major steps: (i)

computation of the ridge directional field, (ii) finding the singularities in the directional image, and (iii) classification of the fingerprint based on the detected singular points. Type and location of a singular point are determined using the Poincaré index technique. Figure 8.1 shows the steps in the classification technique in [5].

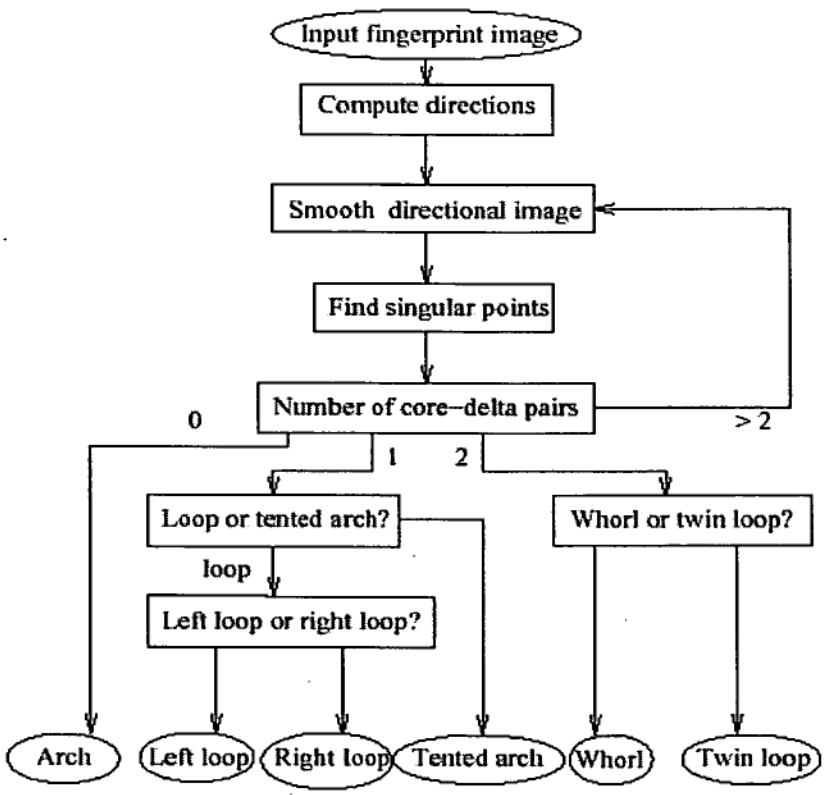


Figure 8.1: Block diagram showing the steps in fingerprint classification technique in [5].

Once the singular points have been located and identified as cores and deltas, the technique classifies the fingerprint image based on the number and location of the singular points according to the rules described in Figure 8.1. However, these features are not sufficient for distinguishing a loop from a tented arch, or twin loop from a whorl. A tented arch can be distinguished from a loop by considering the core-delta axis. In the tented arch, this axis's orientation is along the local direction of the ridges, while in a loop the axis intersects local direction transversally (see Figure 8.2).

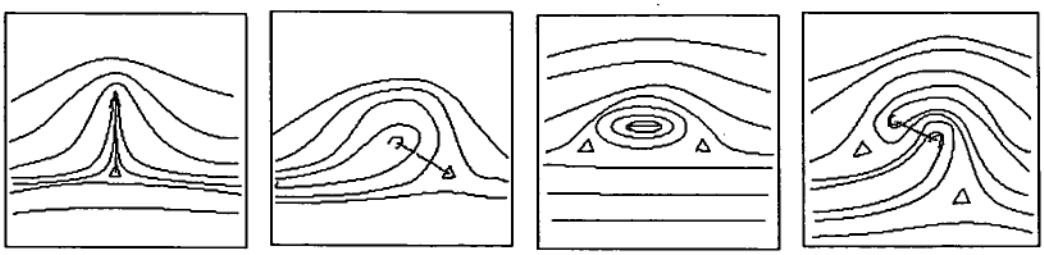


Figure 8.2: Distinguishing a tented arch from a loop and a whorl from twin loops [5,28].

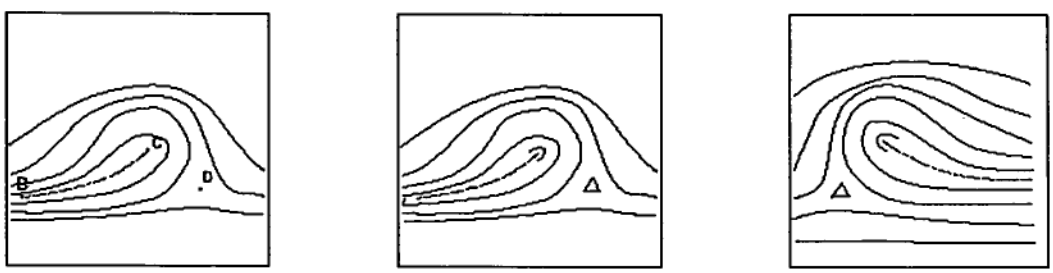


Figure 8.3: Distinguishing a left loop from a right loop [5,28].

The same technique can be used to distinguish a whorl from a twin loop. In a whorl, the core-to-core axis runs almost parallel to the local ridges, while in a twin loop this axis intersects local ridges (see Figure 8.2). The left loop and right loop can be classified by starting to trace from a core point and moving along the direction of the local ridges. The delta point remains to the left in a left loop and to the right in a right loop (see Figure 8.3).

8.2.2 Singular Point Approach 2

This section describes another approach to fingerprint classification based on both singularities and traced pseudoridge analysis [19]. The Poincaré index technique is used both for detecting the location and for determining the type of singular points. Fingerprints are classified into only five types by this technique. But in some cases, tented arch and plain arch are combined into one single class to avoid the difficulty of distinguishing them, and therefore to achieve higher classification accuracy. The main feature of the approach in [19] is the use of pseudoridges in the rules for fingerprint classification. Figure 8.4 shows the two traced pseudoridges. A pseudoridge is found by using three steps: (i) Starting from the core point, trace the

orientation in two opposite directions with a predefined step size, e.g. two or three pixels, until it reaches the window boundary, or until the trace reaches a predefined length, (ii) Out of the two resulted traces, select the one that consistently conforms to the left-hand rule, and (iii) Smooth the selected trace and determine the turning points. The technique is therefore computationally expensive and the tracing of a pseudoridge is very sensitive to the quality of the image.

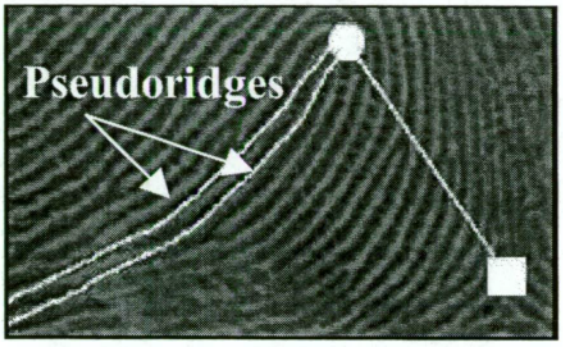


Figure 8.4: Traced pseudoridge superimposed on the fingerprint image [19].

After all singular points and their pseudoridges have been determined, the fingerprint image is classified based on type and location of singular points and the orientation behaviour of pseudoridges. The classification rules have three steps: (i) a fingerprint with no core and no delta is classified as a plain arch, (ii) a fingerprint with two cores is classified as a whorl, (iii) a fingerprint with one core, however, is classified using the shape of the pseudoridge and its turning points as illustrated in Figure 8.5.

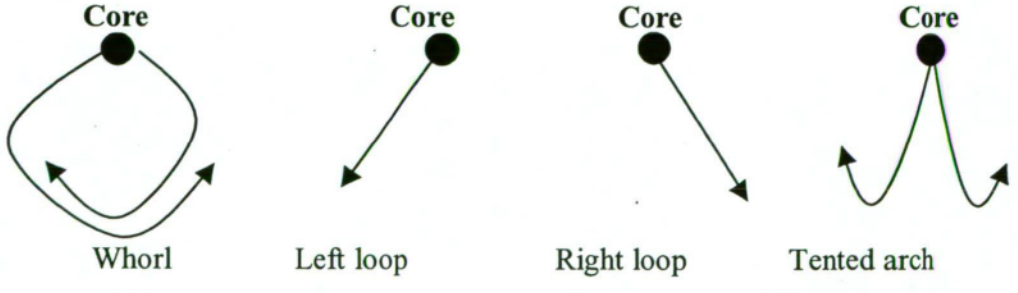


Figure 8.5: Examples of pseudoridges topology in rules for classification.

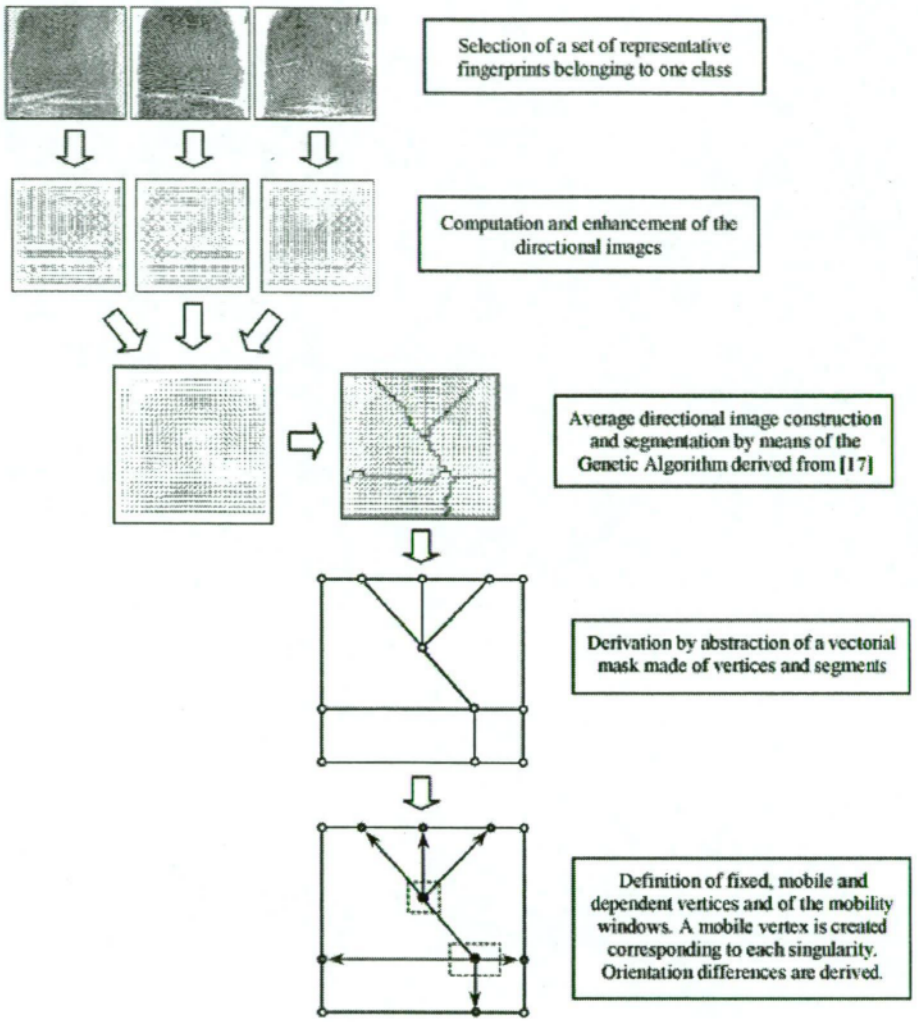


Figure 8.6: Steps for generating prototype masks [14].

8.2.3 Structural Approach

The main idea in the structural approach is to partition the fingerprint into regular-shaped regions of approximately homogeneous orientation so that a best fit relational graph can be found to characterise the macro-features of the fingerprint. A 'guided' dynamic clustering algorithm is used to optimised the segmentation. The relational graph is built by creating a node for each region and an arc for each pair of adjacent nodes. The segmentation employs a set of dynamic guiding masks or prototypes, directly derived from the most common fingerprint classes. The prototype mask for a class is created using in the following steps (see Figure 8.6):

(i) Select a set of representative fingerprints belonging to the chosen class.

(ii) Compute and enhance their directional images.

(iii) Construct and segment the average directional image by means of the Genetic Algorithm.

(iv) Derive by abstraction a vectorial mask made of vertices and segments.

(v) Define fixed, mobile and dependent vertices and the mobility windows. A mobile vertex is created corresponding to each singularity.

The directional image can be computed using ridge orientation estimation in Section 3.2 or the technique used in this approach in [14]. The prototype masks and directional images are used for classifying fingerprints using a neural network or a statistical classifier of exclusive type.

8.3 Classification using Singular Points and their Principal Axes

8.3.1 Determination of Principal Axes

In our technique [6], the ridge orientation field of a fingerprint is transformed into a *ridge flow map* for quickly determining the position of singular points (SPs). These SPs are then classified as an ordinary point, core or delta by calculating the corresponding Poincaré index. Principal axes associated with a singular point are defined as the axes radiated from the point and having the dominant directions of the ridge flow in the immediate neighborhood of the singular point. The directional relationship between the principal axes and the axes joining the singular points, i.e. core-to-core and core-to-delta axes, effectively characterises the global structure of the fingerprint. Principal axes are therefore very useful for distinguishing classes having the same number and type of singular points.

After having determined the location and type of all the singular points, we next calculate the dominant ridge orientation in the immediate vicinity of the cores. The principal axis associated with a core is defined as the line radiated from the core in the dominant ridge direction as shown by white lines in Figure 8.7. A core may have two principal axes as in the case of a tented arch.

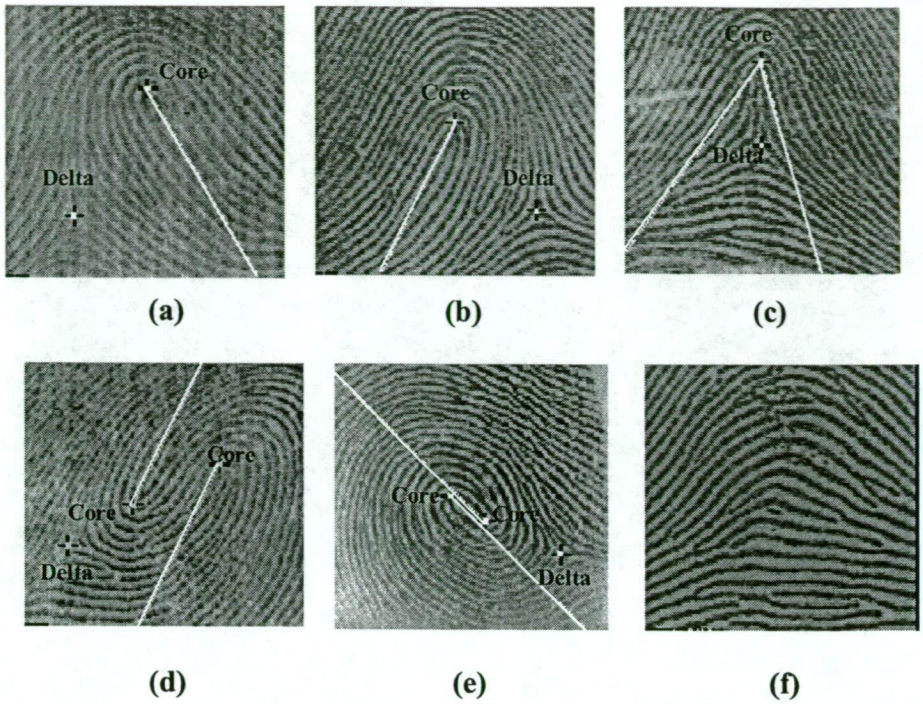


Figure 8.7: Singular points and principal axes (white line) of six classes of fingerprints: (a) and (b) loops, (c) tented arch, (d) twin loop, (e) whorl and (f) plain arch.

8.3.2 Rules for Classification

We then classify a fingerprint image based on the number and the location of its singular points, plus the relative direction (i.e. angle θ) of the principal axes (in white lines) with respect to the core-to-core and core-to-delta axes (in black lines) in Figure 8.8. The classification rules are summarised in Figure 8.9. A loop contains one core-delta pair and one associated principal axis. A tented arch also contains one core-delta pair but has two principal axes, one on each side of the core-delta axis. A twin loop and a whorl each have two cores and two principal axes. A plain arch fingerprint contains no cores or deltas, nor principal axes.

As mentioned in the introduction of this chapter, current techniques [5,14,19,28] classify fingerprints into only five classes with twin loop and whorl being combined into one class. Also in practical situations, delta points of whorl and twin loop are usually located in the outer region of a fingerprint, and consequently are often missed out from the fingerprint image. The fingerprint, in this situation, would be rejected by current classification techniques. In our technique however, since the pairing of core and delta is assumed, the missing deltas do not handicap the proposed technique. The

principal axes associated with the cores are used instead to form the classification rules for twin loops and whorls. Figure 8.8 shows a considerable difference between a whorl and a twin loop, and between a tented arch and a loop.

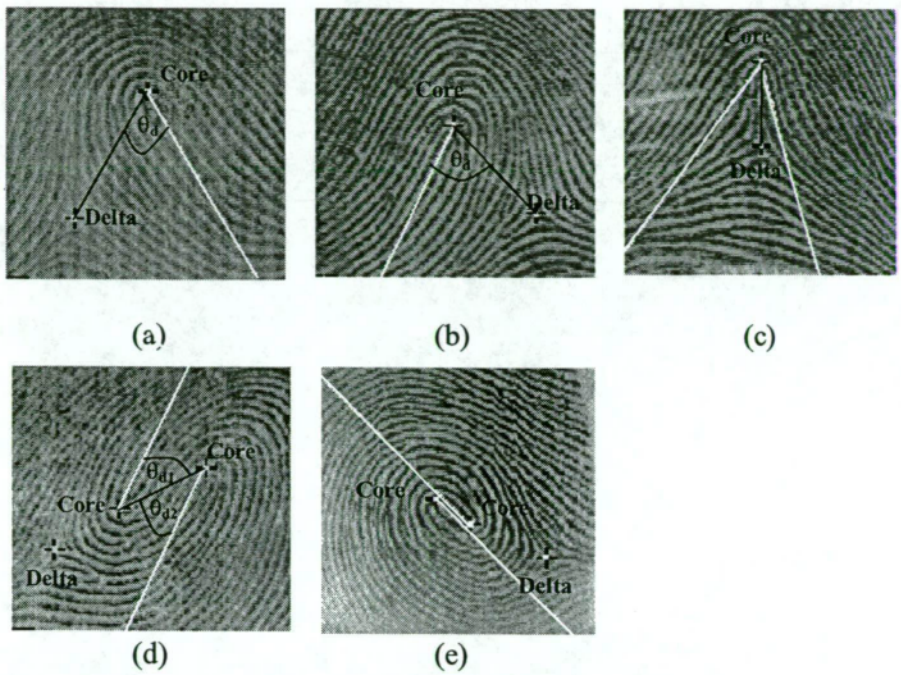


Figure 8.8: Definition of the relative direction of principal axes (a) right loop, (b) left loop, (c) tented arch, (d) twin loop, and (e) whorl.

Let θ_d be the relative angle between the principal axis and the core-to-core axis. If θ_d is more than a threshold on the right or on the left, then the fingerprint is classified as right loop or left loop, respectively. The same method can be used to distinguish between a whorl and a twin loop. If both θ_{d1} and θ_{d2} are less than a threshold, then the fingerprint is classified as a whorl, otherwise it is a twin loop

8.3.3 Case of 'Fat' Loops

A 'fat' loop, as illustrated in Figure 8.10 (a), may have two dominant ridge orientations in the immediate vicinity of its core, hence two principal axes. For the proposed classification algorithm to work, it is necessary to retain only the correct axis.

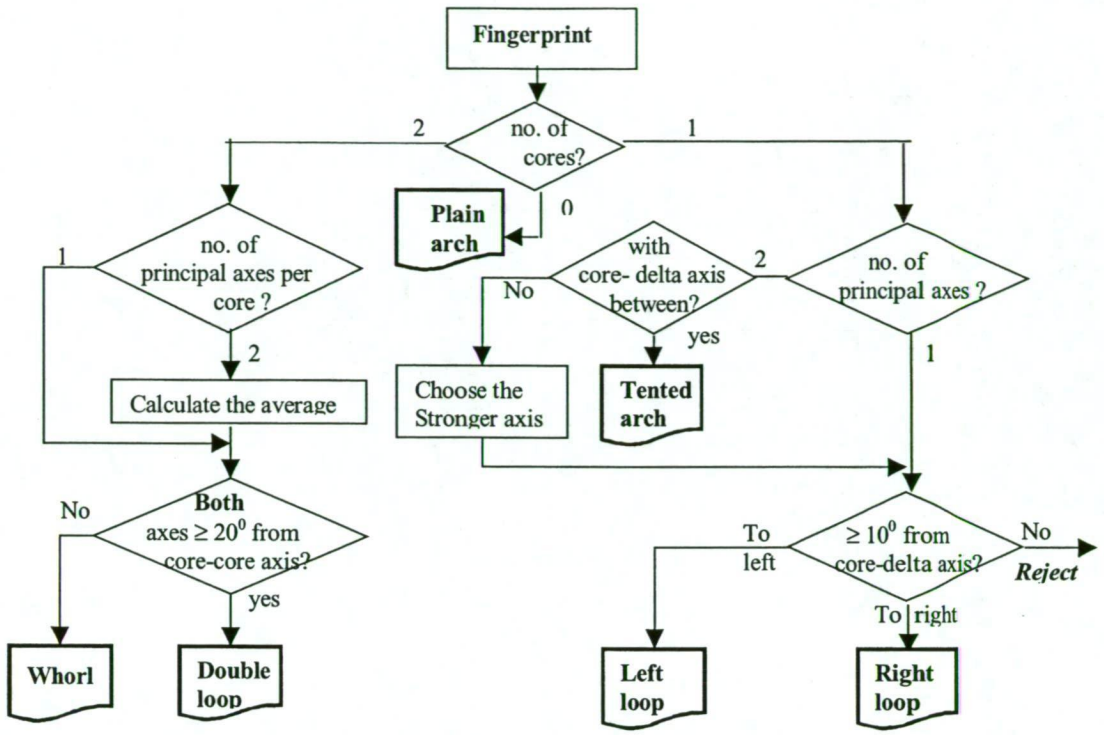


Figure 8.9: Proposed rule-based fingerprint classification algorithm using singular points and their principal axes

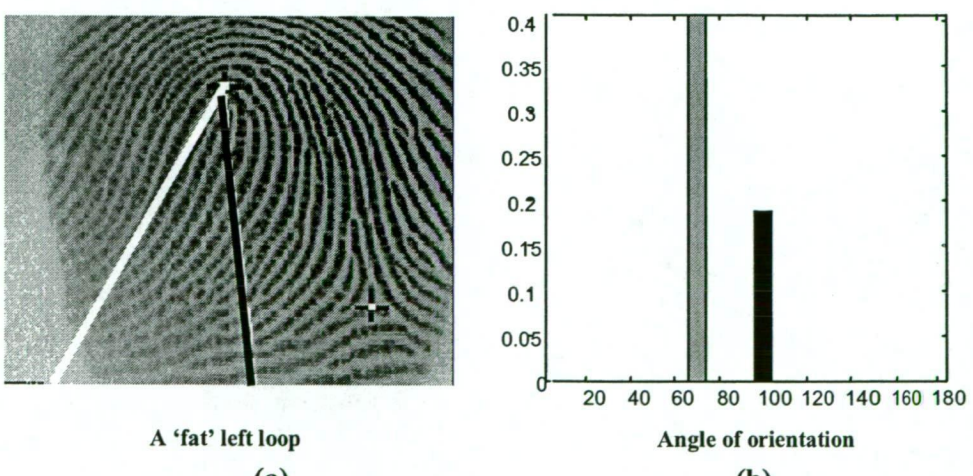


Figure 8.10: Example of a left loop having two dominant ridge orientations in the vicinity of its core point, (a) a 'fat' left loop, and (b) histogram of ridge orientation in the vicinity of the core point.

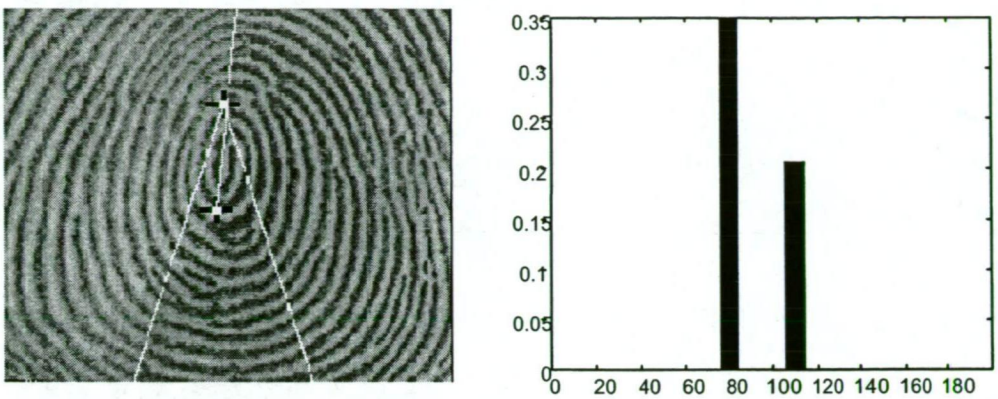


Figure 8.11: Example of two principal axes from the upper core of a whorl, (a) whorl, and (b) histogram of ridge orientation in the vicinity of the upper core point.

From the histogram in Figure 8.10 (b), it is obvious that the orientation shown in black is the weaker of the two principal axes, and therefore has to be eliminated. On the other hand, it is expected that each core of a whorl has two principal axes, one on each side of the core-to-core axis. As shown in Figure 8.11 (b), the two principal axes from a core in a whorl typically have more comparable strengths.

It is obvious that the two principal axes of a core in a whorl radiate out fairly symmetrically from the core. Consequently, the average direction of the two is expected to run very close to the core-to-core axis of the whorl as illustrated in Figure 8.8(e). The proposed classification rule is that if both average axes (directions) having θ_{d1} and θ_{d2} less than a threshold, then the fingerprint image is classified as a whorl.

8.4 Experimental Results and Conclusions

A simple technique has been successfully presented for the detection of singular points and their associated principal axes on a fingerprint. The accurate detection of these two global characteristic structures of fingerprints has allowed the author to propose a classification algorithm that is an improvement compared to current ruled-based techniques. For a small database of only 157 fingerprints being available to the

author, the classification result is recorded in Table 8.1 with an overall correct classification rate of 91.33%. An outcome is rejected when singular points are not detected in core-delta pairs. The separation of a whorl from a twin loop represents by far the most challenging task. Previous researchers, e.g.[5,28], did not tackle this difficulty but combined whorl and twin loop into a single class.

Table 8.1: Result of classification

True class / Assigned	**Plain arch (19)**	**Tented arch (11)**	**Whorl (34)**	**Left loop (47)**	**Right loop (34)**	Twin loop (12)
Plain arch	17	1	0	1	1	0
Tented arch	1	8	0	0	1	0
Whorl	0	0	29	0	0	2
Left loop	0	0	0	43	0	0
Right loop	0	1	0	0	30	0
Twin loop	0	0	5	0	0	10
Reject	1	1	0	3	2	0

Chapter Nine: Conclusion and Future Investigation

The primary focus of the work in this thesis is on fingerprint classification. Fingerprint identification and verification are not within the scope of our study due to the short time limit of the thesis. We have proposed a number of new techniques and algorithms in various places in the three main phases of fingerprint classification: enhancement and filtering, determination of singular points and principal axes, and classification.

We have implemented current techniques and the proposed technique for fingerprint enhancement and compared their performance using the *goodness index* rather than subjective observation. This is because it is not easy for lay observers to visually spot the difference between two enhanced images. The goodness index indirectly assesses the improvement in fingerprint quality in terms of the accuracy of the minutiae extraction. Strictly speaking, the performances of two enhancement algorithms cannot be compared fairly using the goodness index because this measure is dependent on the ridge structure. For example, a uniform ridge structure can be enhanced by a simple fixed-frequency Gabor filter as well as by a Gabor filter that is adaptively tuned to the local ridge frequency, as proposed in this thesis. For fingerprints with less uniform ridge structure, the proposed technique has proved to give better enhancement than current techniques using fixed frequency Gabor filters. We have proposed a pre-calculated 2-D Gabor filter array from which 2-D interpolation is used to obtain the filtered image at any ridge orientation and any ridge frequency estimated for the local region.

Since the determination of singular points relies heavily on estimated local ridge orientation, we have proposed to double the (linear) estimation resolution by calculating the orientation field of the fingerprint on a quincunx grid. Experimental results have shown that our technique has the ability to accurately detect not only the singular points but also the extra points of maximum orientation variance in plain arches (having no core and no delta). These detected extra points should be useful for the alignment of two plain arches for minutia matching. Furthermore, experiments have also shown that the proposed technique for singular point detection is both scale

and rotation invariant to within the rounding accuracy of the (x,y) coordinates of an image block containing the singular point. In particular, under a rotation of more than about 20°, the ridge flow pattern is unavoidably distorted by the co-ordinate transformation process and may result in significant error in the location of singular points.

The principal axis associated with a singular point is defined as the axis radiated from the point and having the dominant direction of the ridge flow near the singular point. Singular points and their associated principal axes are used to formulate rules for fingerprint classification. A core of a loop may have two comparable dominant principal axes giving confusion in distinguishing a loop from a tented arch. We have proposed a masking technique using an orientation histogram to mask out the unwanted axis. An orientation mask is designed for excluding the inner region around a singular point and suppressing non-radial orientations.

By incorporating all previous findings about the characteristics of local ridge patterns such as singular points and their associated principal axes, as well as global ridge flow patterns, we have successfully proposed a rule-based fingerprint classification algorithm. Experimental results have highlighted the fact that the classification of a whorl from a twin loop remains a challenging task. Our technique can classify fingerprints into six classes with high accuracy. Current techniques avoid the above challenge by combining isolated whorl and twin loop into one single class.

Future investigations to be carried out include the following issues that we have identified:

A more accurate technique for determining locations of singular points and associated principal axes of fingerprint images rotated by a large angle.

Most fingerprints acquired from scanners have lost their delta points because these points are usually located on the lower edge of the finger and therefore are not covered by the scanner.

Appendix A: Examples of Experimental Results

The two challenging patterns with plotted principal axes as shown in Figure A.1 and A.2.

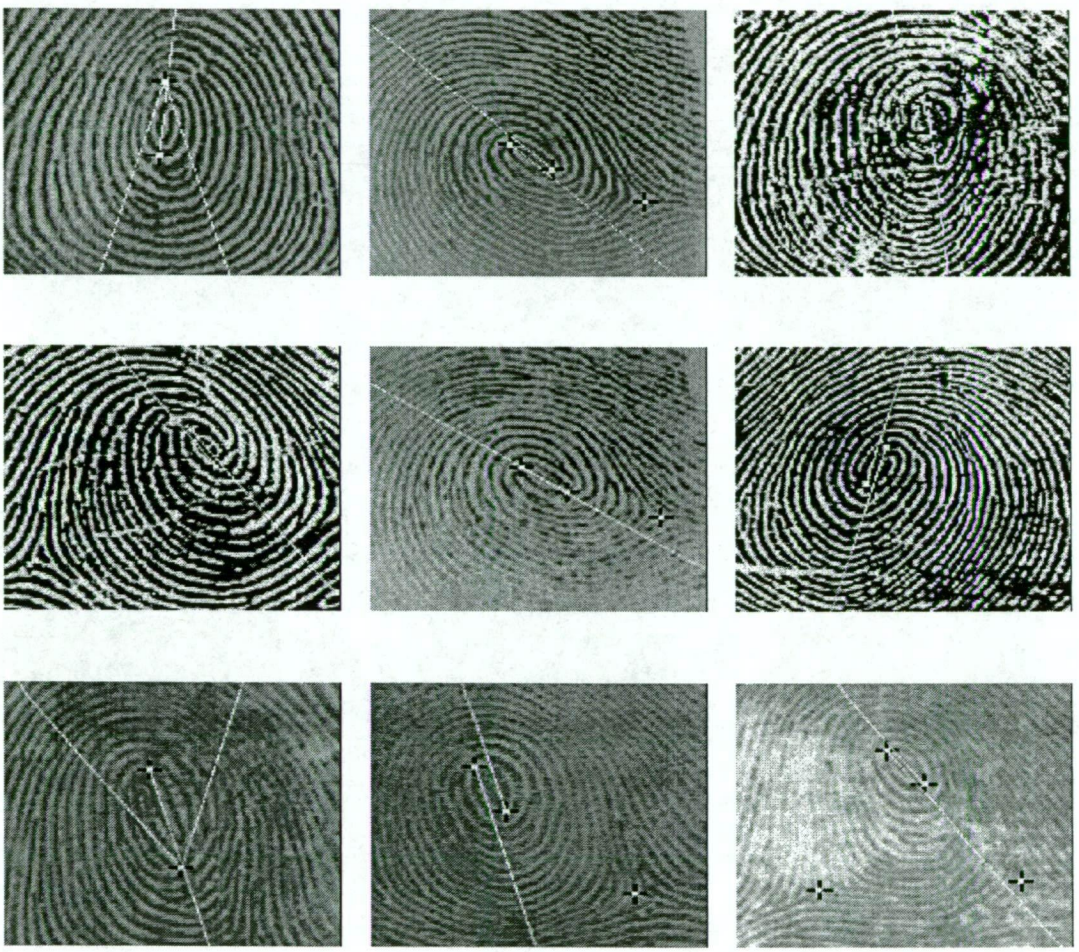


Figure A.1: Whorl with plotted principal axes.

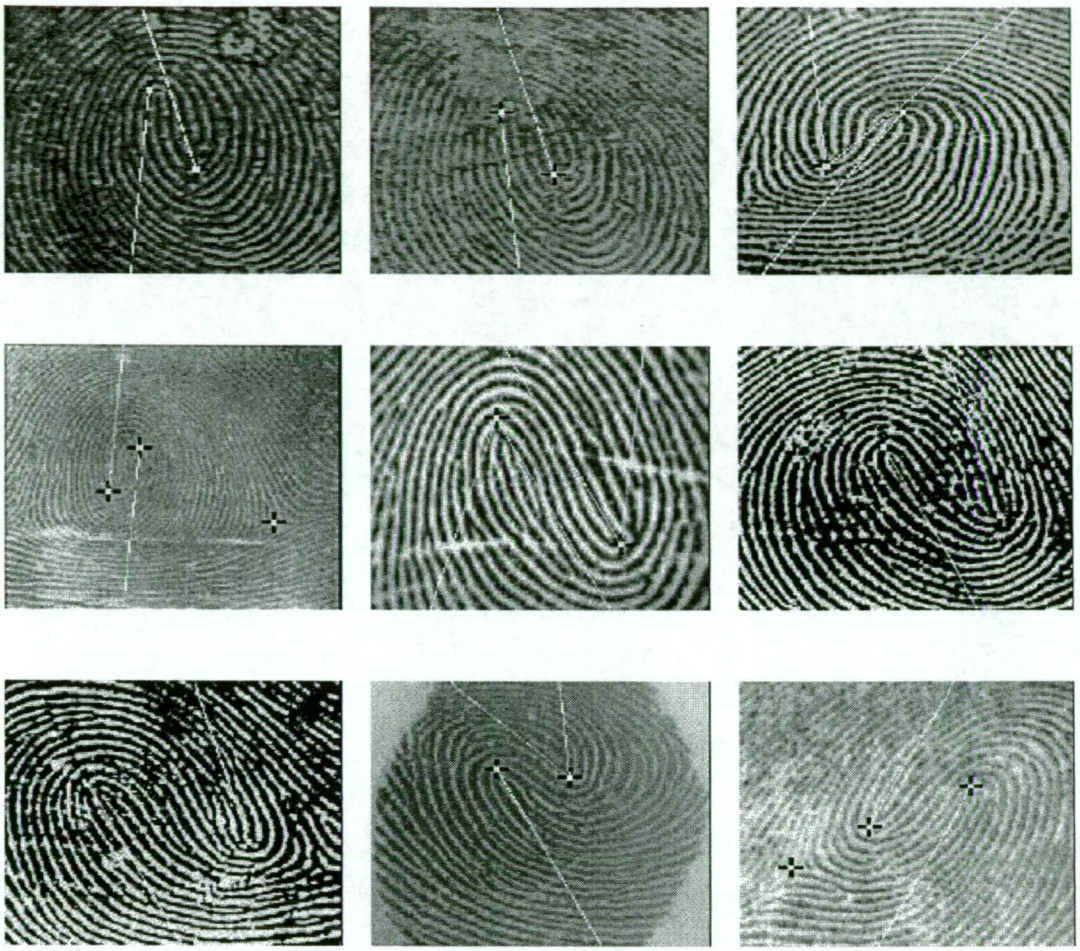


Figure A.2: Twin loop with plotted principal axes.

The fingerprint patterns as tented arch, left loop and right loop with plotted principal axes are shown in Figure A.3, A.4 and A.5 respectively.

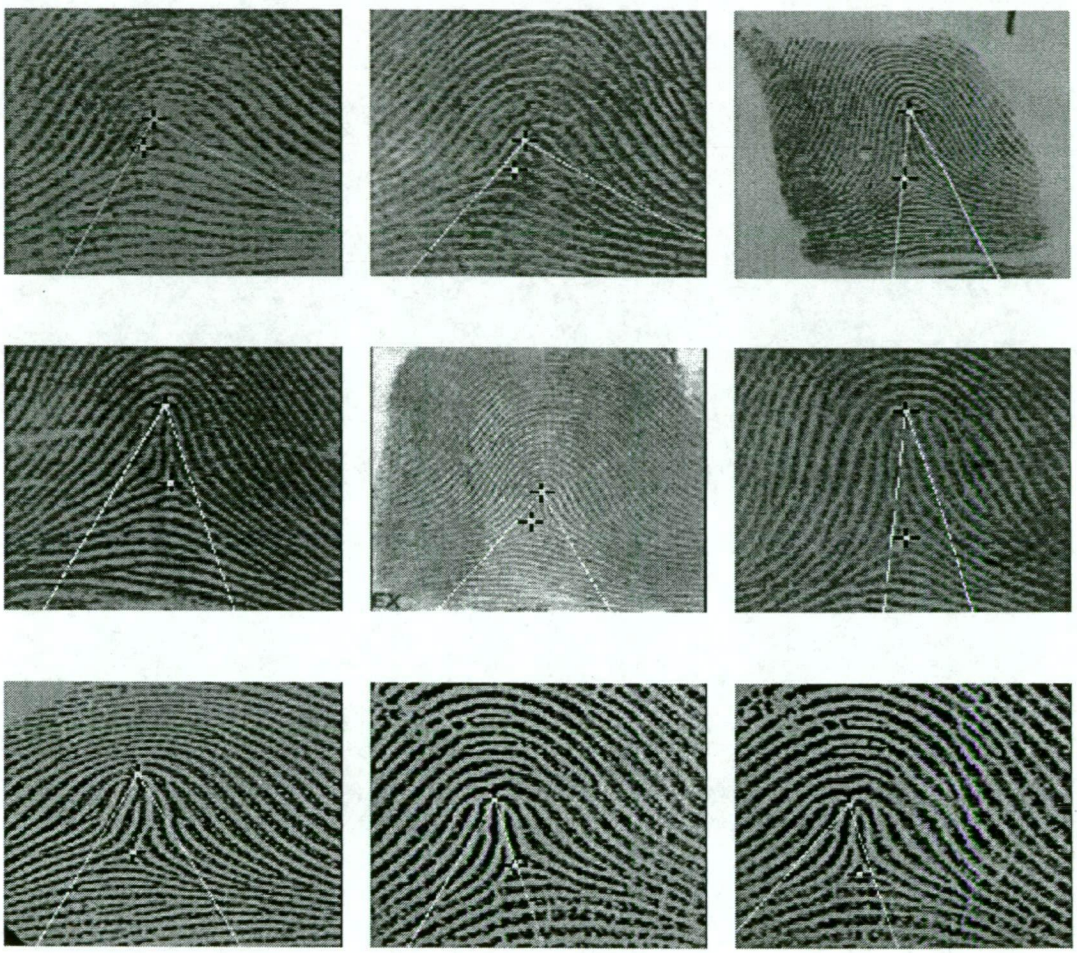


Figure A.3: Tented arch with plotted principal axes.

Figure A.4: Right loop with plotted principal axes.

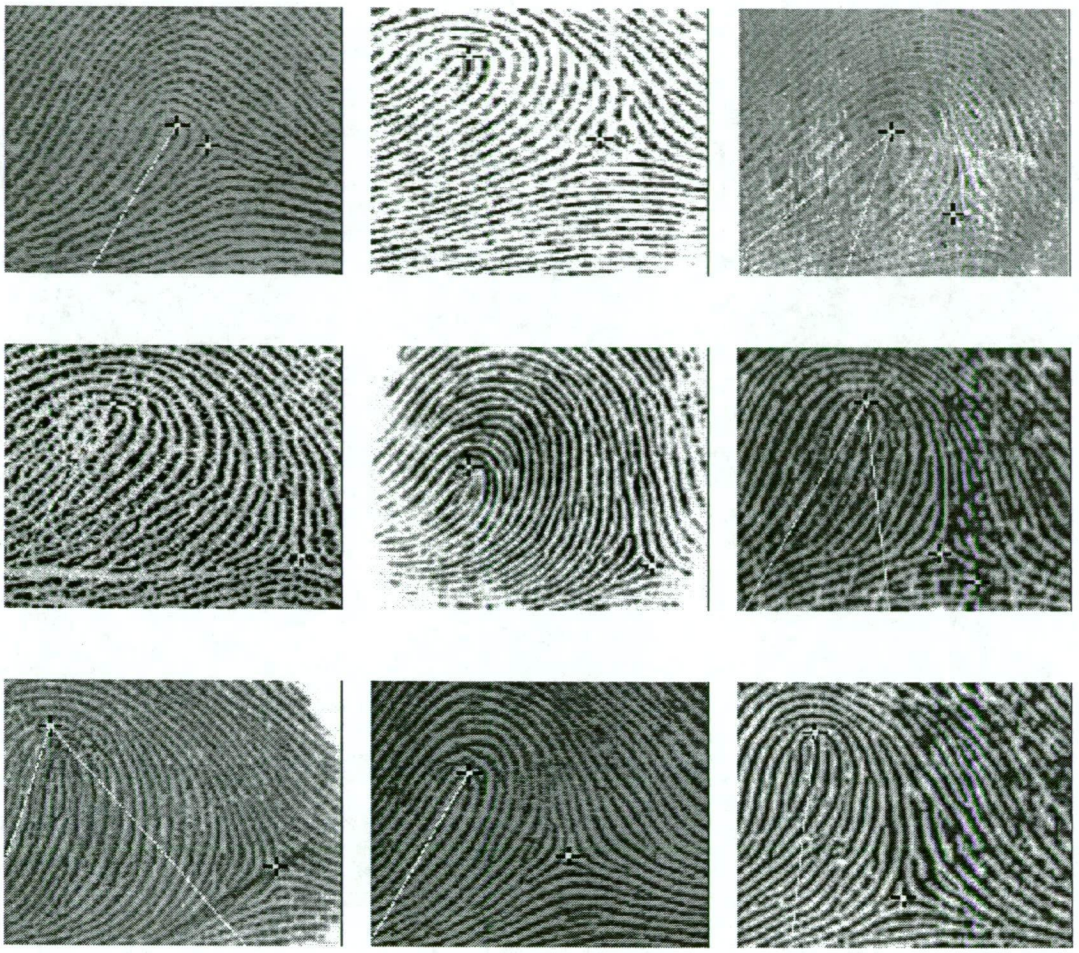


Figure A.5: Left loop with plotted principal axes.

References

[1] Ed German, "The History of Fingerprints", available at: http://onin.com/fp/fphistory.html

[2] Anil K. Jain, Arun Ross and Salil Prabhakar, "An Introduction to Biometric Recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, Special Issue on Image- and Video-Based Biometrics, August 2003

[3] A. K. J. L.Hong, S. Pankanti, and R. Bolle, "Fingerprint Enhancement," *Proc. First IEEE WACV*, Sarasota, Fla, 1996.

[4] C. Klimanee and D.T. Nguyen, "On the Design of 2-D Gabor Filtering of Fingerprint Images," *Proceedings IEEE International Consumer Communications and Networking Conference CCNC'04*, Caesar's Palace, Las Vegas, Nevada USA, 6-9 January 2004.

[5] K. Karu and A.K. Jain, "Fingerprint Classification," *Pattern Regcognition*, Vol. 29, No. 3, pp. 389-404, 1996.

[6] C. Klimanee and D.T. Nguyen, "Classification of Fingerprints using Singular Points and Principal Axes," *Submitted to IEEE International Conference in Image Processing ICIP'04,* Singapore November 2004.

[7] Fr. Don Matthys, S.J. ,"Contrast Stretching", available at: http://academic.mu.edu/phys/matthysd/web226/Lab01.htm

[8] L. Hong, Y. Wan, and A.J. Jain, "Fingerprint Image Enhancement: Algorithm and Performance Evaluation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 8, pp. 777-789, August 1998.

[9] A. Ravishanka Rao: *A Taxonomy for Texture Description and Identification*, Springer- Verlag, New York, 1990

[10] D. Maio, D. Maltoni, R. Cappelli, J. L. Wayman and A. K. Jain ," Fingerprint Verification Competition", available at: http://bias.csr.unibo.it/fvc2002/default.asp

[11] I.S. Gradshteyn and I.W. Ryzhik , *Table of Integrals, Series and Products,* Academic Press, New York, 1965

[12] A.M. Baze, G.T.B. Verwaiijen, S.H. Gerez, L.P.J. Veelenturf, and B.J. van der Zwaag, "A correlation-based fingerprint verification system," *Proc. ProRISC2000 Workshop on Circuits, Systems and Signal Processing*, Veldhoven, Netherlands, November 2000.

[13] Rafael C. gonzalez and Richard E. Woods: *Digital Image Processing*, Prentice-Hall, New Jersey 2002.

[14] R. Cappelli, A. Lumini and D. Maio, "Fingerprint Classification by Directional Image Partitioning," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 21, no. 5, pp. 402-421, May 1999.

[15] A.V. Trinh and D.T. Nguyen, "Fingerprint Classification based on Extracted Singular Points," *Proceedings 7th International Symposium on Digital Signal Processing and Communication Systems DSPCS'2003*, Gold Coast, Australia, 8-11 December 2003.

[16] M. Kawagoe and A. Tojo, "Fingerprint Pattern Classification," *Pattern Regcognition*, Vol. 17, No. 3, pp. 295-303, 1984.

[17] R. Rosen, *Dynamical System Theory in Biology, Volume I: Stability Theory and its Applications*, John Wiley, New York, 1970.

[18] A.Ross, A.K. Jain and J.Reisman, "A Hybrid Fingerprint Matcher," *Pattern Recognition*, Vol.36, pp.161-1673, 2003.

[19] K. H. a. H. Y. Qinzhi Zhang, "Fingerprint Classification Based on Extraction and Analysis of Singularities and Pseudoridges," *Proceedings of the Pan-Sydney area workshop on Visual information processing, Sydney Australia*, pp.83-87, 2001.

[20] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, "Handbook of Fingerprint Recognition," Springer Verlag, June 2003.

[21] S. Prabhakar, A. K. Jain, and S. Pankanti, "Learning Fingerprint Minutiae Location and Type," *Pattern Recognition*, Vol. 36, No. 8, pp. 1847-1857, 2003.

[22] S. Pankanti, S. Prabhakar, and A. K. Jain, "On the Individuality of Fingerprints," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 24, No. 8, pp. 1010-1025, 2002. A shorter version also appears in *Fingerprint World*, pp. 150-159, July 2002.

[23] A. K. Jain and S. Pankanti, "Automated Fingerprint Identification and Imaging Systems," *Advances in Fingerprint Technology*, 2nd Ed. (H. C. Lee and R. E. Gaensslen), CRC Press, 2001.

[24] Salil Prabhakar, "Fingerprint Classification and Matching Using a Filterbank," Computer Science & Engineering of Michigan State University, *Ph.D. Thesis*, 2001.

[25] A. K. Jain, S. Prabhakar and S. Pankanti, "Matching and Classification: A Case Study in Fingerprint Domain," *Proc. INSA-A (Indian National Science Academy)*, Vol. 67, A, No. 2, pp. 223-241, March 2001.

[26] S. Prabhakar, J. Wang, A. K. Jain, S. Pankanti, and R. Bolle, "Minutiae Verification and Classification for Fingerprint Matching," *Proc. 15th International Conference Pattern Recognition*, Vol. 1, pp. 25-29, Barcelona, September 3-8, 2000.

[27] A. K. Jain, S. Prabhakar, L. Hong and S. Pankanti, "FingerCode: A Filterbank for Fingerprint Representation and Matching," *Proc. IEEE Conference on CVPR*, Colorado, Vol. 2, pp. 187-193, June 23-25, 1999.

[28] L. Hong and A. K. Jain, "Classification of Fingerprint Images," *Proceedings of 11th Scandinavian Conference on Image Analysis*, June 7-11, Kangerlussuaq, Greenland, 1999.

[29] A. K. Jain and S. Pankanti, "Biometrics Systems: Anatomy of Performance," *IEICE Trans. Fundamentals*, Special issue on biometrics, Vol. E84-D, No. 7, July 2001.

[30] Yongkang Ye and Huang Ju, "DNA Electrochemical Behaviours Recognition and Sensing by Combining with PCR Technique," Molecular Diversity Preservation International *(MDPI) Trans. Sensors*, pp. 128-145, No. 3, 28 July 2003.

[31] Bahadir K. Gunturk, Aziz U. Batur, Yucel Altunbasak, Monson H. Hayes and Russell M. Mersereau, "Eigenface-Domain Super-Resolution for Face Recognition," *IEEE Trans. Image Processing*, Vol. 12, No. 5, May 2003.

[32] A. Kumar, D.C.M. Wong, H.C. Shen, A.K. Jain, "Personal Verification using Palmprint and Hand Geometry Biometrics," *Proceedings of the fourth International Conference on audio- and video-based biometric personal authentication*, June 2003

[33] Jafar M. H. Ali Aboul Ella Hassanien, "An Iris Recognition System to Enhance E-security Environment Based on Wavelet Theory," *AMO - Advanced Modeling and Optimization,* Volume 5, Number 2, 2003

[34] Nong Ye and Xiangyang Li, "A Scalable Clustering Technique for Intrusion Signature Recognition," *IEEE Trans. Workshop on Information Assurance and Security* , United States Military Academy, West Point, NY, 5-6 June, 2001.

[35] H. Baltzakisa and N. Papamarkosb, "A new signature verification technique based on a two-stage neural network classifier," *Engineering Applications of Artificial Intelligence*, Vol. 14, pp.95-103, 2001